

Większa część obrazów używanych przez Blender pochodzi bezpośrednio z dwóch plików Inkscape:

- **holes.png**: to wyeksportowane z pliku *skin.svg* warstwy **Holes + BkgWhite**;
- **nor\_details.png**: to wyeksportowane z pliku *skin.svg* warstwy **Result:Nor-Details + BkgGrey** (nieprzejrzyste w 100%);
- **nor\_blur.png**: to wyeksportowane z pliku *skin.svg* warstwy **Result:Nor-Blur + BkgWhite**. Opcjonalnie można ten obraz przekształcić z *Bump Map* w *Normal Map* za pomocą wtyczki Gimpa (por. str. 409);
- **decals.png**: to wyeksportowane z pliku *decals.svg* warstwy **Color-Insingnia + Color-Technical** (bez żadnego tła);

Dotyczy to także dwóch tekstur pomocniczych (nie narysowałem ich na poprzedniej stronie, by nie komplikować schematu):

- **ambient.png**: to wyeksportowane z pliku *decals.svg* warstwy **Ambient + Bkg-Black + Bkg-White** (służy do lokalnego tłumienia lub wzmacniania efektu *Ambient Occlusion* — por. str. 403);
- **cusps.png**: to wyeksportowane z pliku *decals.svg* warstwy **Cusps + Bkg-White** (służy do tłumienia niepożądanych odbłyśków — por. str. 404);

Oprócz tego z pliku Inkscape *skin.svg* eksportowane są dwa obrazy „do dalszej obróbki” w GIMP:

- **shadows.png**: to wyeksportowane z pliku *skin.svg* warstwy **Result:Ref-Other + Result:Ref-Grunge + Result:Ref-Blur + BkgGrey** (nieprzejrzyste w 60%) + **BkgWhite**. Obraz należy wstawić do pliku GIMP *skin.xcf* jako warstwę **Ref-shadows.png** (por. Rysunek 6.13.1);
- **screws.png**: to wyeksportowana z pliku *skin.svg* warstwa **Screws**, bez żadnego tła. Ten obraz można opcjonalnie wstawić do *skin.xcf* jako warstwę **Color-screws.png**. Służy do rozjaśnienia łbów śrub na teksturze barwy;

Ostatecznie, z pliku Gimpa uzyskujemy obrazy dwóch istotnych tekstur:

- **ref.png**: to wyeksportowane ze *skin.xcf* złożenie wszystkich warstw o przedrostku **Ref\*** (**Ref-Dirt, Ref-Smoke, Ref-shadows.png**);
- **color.png**: to wyeksportowane ze *skin.xcf* złożenie wszystkich warstw o przedrostku **Color\***;

- Podczas prac nad detalami samolotu na pewno będzie trzeba uzupełnić teksturę poszycia o jakieś śruby, nity, czy łączenia blach. Oznacza to, że jeszcze co najmniej kilka razy będziesz musiał wygenerować nowe wersje plików obrazów dla Blendera.

Zmiany obrazów tekstur należy nanieść w odpowiednich warstwach „szczegółowych” pliku *skin.svg* (**Screws, Rivets, PanelIV, PanelH, ...**) wewnątrz umieszczonych na nich grup (**#Screws, #Rivets, #PanelIV, #PanelH, ...**). Warstwy przygotowane w *skin.svg* do eksportu (**Result:\***) zawierają referencje do tych grup, więc są natychmiast aktualizowane. Po wprowadzeniu zmian w Inkscape należy ponownie „przejsć” przez schemat, przedstawiony przez Rysunek 6.13.1:

1. wygenerować pięć plików z Inkscape (*holes.png, screws.png, shadows.png, nor\_details.png, nor\_blur.png*);
2. włączyć do GIMP pliki *shadows.png* i *screws.png* (na odpowiednie warstwy). To tymczasowe pliki, które po wczytaniu można usunąć;
3. wygenerować ze *skin.xcf* nowe wersje *color.png* i *ref.png*;

Oczywiście, jeżeli niczego nie zmieniałeś w układzie „dziur” na warstwie **Holes**, możesz nie generować pliku *holes.png*. To samo dotyczy *screws.png*. Pozostałe trzy pliki, wyliczone w pkt. 1, są złożeniem wielu grup, i na wszelki wypadek generuj je po każdej zmianie!

## **Rozdział 7. Pozostałe detale samolotu**

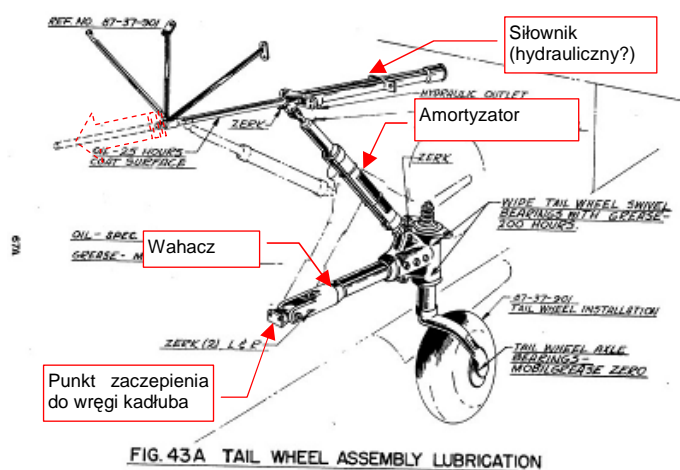
W tym rozdziale zakończymy modelowanie samolotu. Powstanie tu mnóstwo drobnych części, które — w większości przypadków — nie będą rozwijane i teksturowane. Zamiast tego do ich pokrycia użyjemy prostszych, jednorodnych materiałów, opartych o tekstury proceduralne. Nie będę w tym rozdziale opisywał każdego detalu. Skupię się raczej na tych, które wymagają jakiejś nowej, nie omawianej do tej pory, metody wykonania. Zaczniemy, dla nabrania wprawy, od szczegółowego omówienia sposobu tworzenia kółka ogonowego. To prosty mechanizm, więc dobrze jest na nim pokazać, jak „pracować z detalem”. Potem przejdziemy do bardziej złożonych całości: najpierw podwozia głównego, a później — do wnętrza kabiny pilota.

Gdy skończysz kabinę pilota, cała reszta pracy wyda Ci się już łatwa i bardzo prosta!

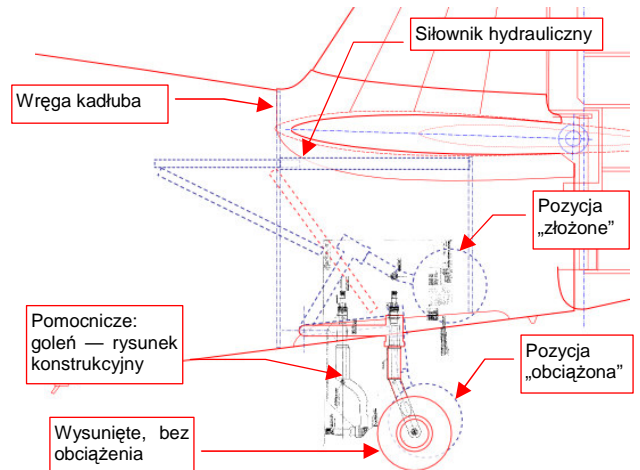
## 7.1 Kółko ogonowe

Szczegóły samolotu zaczniemy od kółka ogonowego. Wybrałem je na początek tego rozdziału, gdyż nie jest zbyt skomplikowane. Jednocześnie wykonanie tego elementu pozwala pokazać kilka technik, które zastosujemy także w innych miejscach modelu.

Nim zabierzesz się za modelowanie podwozia, postaraj się zgromadzić jak najwięcej zdjęć tego elementu. Czasami są to dość skomplikowane mechanizmy! Na szczęście kółko ogonowe P-40 do takich nie należy<sup>1</sup>. Wśród dokumentacji, opublikowanej w Internecie, można znaleźć całkiem przejrzysty rysunek tego zespołu (Rysunek 7.1.1 — w istocie to schemat okresowego smarowania). Dodatkowo znalazłem tam także rysunek fabryczny goleni. Przygotowałem na tej podstawie „szkic wykonawczy” kółka ogonowego, wkomponowany w plany samolotu (Rysunek 7.1.2):

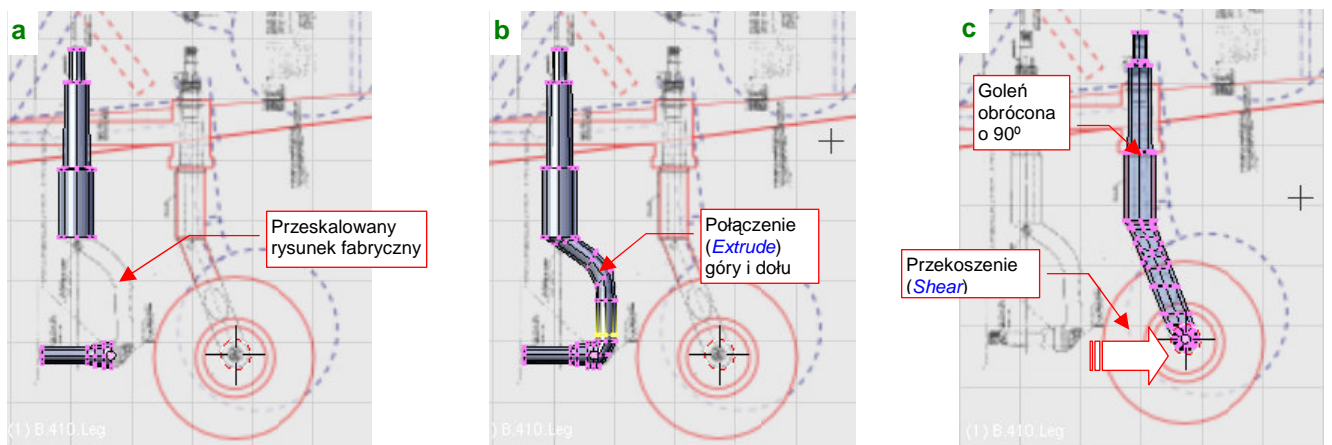


Rysunek 7.1.1 Oryginalny schemat zespołu kółka ogonowego



Rysunek 7.1.2 Rysunki wykonawcze dla modelu

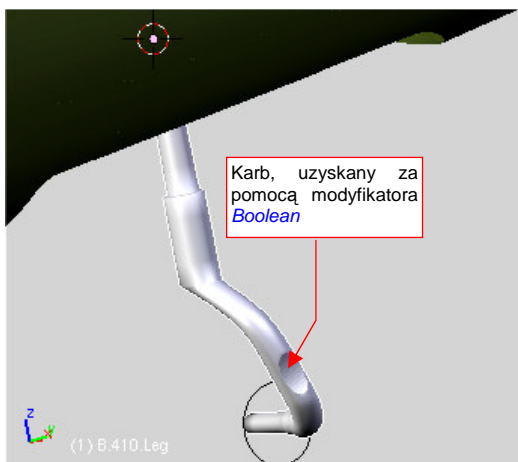
Cały zespół zaczniemy od goleni. W oparciu o rysunek fabryczny, zmniejszony do odpowiedniej skali, uformowałem najpierw dwa elementy cylindryczne: oś koła i górną część goleni (Rysunek 7.1.3a):



Rysunek 7.1.3 Formowanie goleni kółka ogonowego

Połączyłem te dwie „rury” wytłoczeniem, wygiętym jak na rysunku fabrycznym (Rysunek 7.1.3b). Następnie odwróciłem całą goleń o 90°, w położenie docelowe, i przesunąłem (przekoszeniem — *Shear*) oś w położenie zgodne z rzutem z boku (Rysunek 7.1.3c).

<sup>1</sup> Warto zauważyć, że w odróżnieniu od wielu innych konstrukcji z tego okresu (np. Bf-109G, Ła-5) mechanizm chowania kółka ogonowego w P-40 nie sprawiał problemów podczas eksploatacji z lotnisk polowych. Nie zauważyłem nigdzie przypadków blokowania tego kółka przez warsztaty w pozycji „wysunięte” — bardzo częstych sytuacji w myśliwcach Luftwaffe i WWS.



Rysunek 7.1.4 Uformowana goleń (B.410.Leg)

potrafi „z własnej inicjatywy” dorzucić do rezultatu dodatkowe ściany. Trzeba mu wtedy te twórcze podejście wyperswadować, dodając do siatki parę dodatkowych krawędzi. (Chodzi o to, by zmienić jej topologię, uzyskując inny przebieg linii wzajemnego przenikania powłok).

Od razu przypiszmy wykonany element do odpowiedniego materiału. Na zachowanych kolorowych zdjęciach z lat 40-tych XX w. podwozie P-40 ma wyraźny, metaliczny połysk. Nie wygląda to na farbę, tylko na inny sposób pokrycia (np. ocynk<sup>1</sup>). Taka powłoka miałaby w tym miejscu sens. W końcu podwozie jest bardziej narażone na korozję, niż reszta samolotu.

Stworzyłem odpowiedni materiał, który będzie wyglądał jak lekko chropowata stal, pokryta jakąś warstwą zabezpieczającą w kolorze metalu. Szczegóły, jak to zrobić, znajdziesz na str. 989. Przygotowałem materiał tak, jak tam podano, tylko zmniejszyłem wartość *Ref* (panel *Shaders*) z 1.0 do 0.7. Nadałem mu nazwę **B.Steel.Rough**, i przypisałem do siatki **B.410.Leg**. Rysunek 7.1.5 pokazuje, jak wygląda goleń podwozia na próbnym renderze.



Rysunek 7.1.5 Próbnny rendering z materiałem B.Steel.Rough.

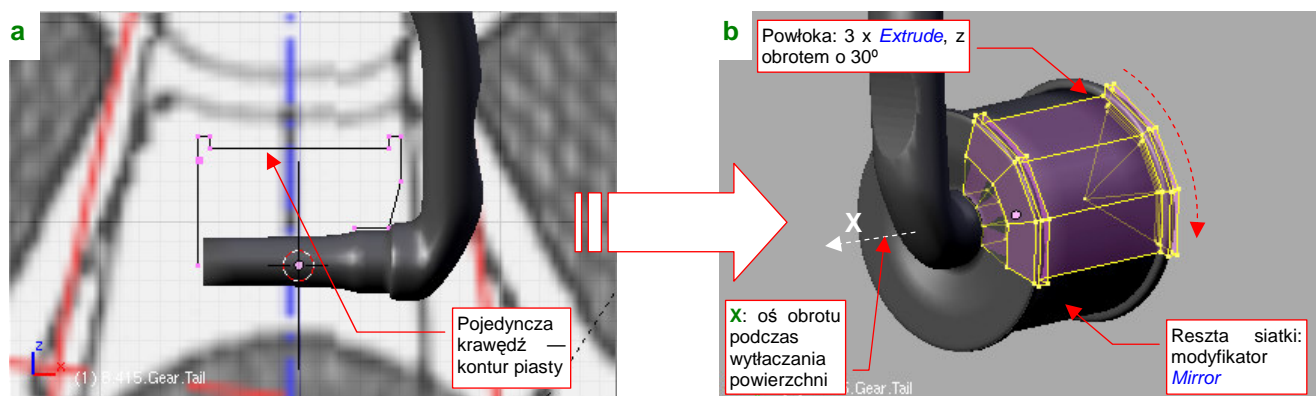
- Dla wszystkich drobnych części samolotu, za wyjątkiem paru elementów kabiny pilota, nie stosuj pracochłonnych rozwinięć w UV i „ręcznie malowanych” tekstur. Na typowych ujęciach i tak będą za małe, by takie detale można było dostrzec.

Zamiast tego, wykorzystaj do pokrycia drobnych części takie materiały jak **B.Steel.Rough**. Użyto w nich co najwyżej tekstur proceduralnych, nałożonych jakimś inną metodą niż pracochłonne UV (np. **Orco**, albo **Nor**).

Oczywiście, jeżeli planujesz robić jakieś ujęcie podwozia z bliska, to wtedy warto się postarać. Trzeba na każdą siatkę nanieść szwy, rozwinąć, sprawdzić „jak leży” testowy obraz na poszczególnych detalach. Potem na obrazie tekstury namalować wszystkie zabrudzenia, plamy oleju, zacieki... Licz się jednak z tym, że zajmie Ci to dzień lub dwa. To poniekąd powtórzenie, tylko w trochę mniejszej skali, tego, co opisywał Rozdział 6. Mówiąc krótko: masa dodatkowej roboty!

<sup>1</sup> Mam na myśli pokrycie elementów stalowych warstwą cynku. To popularna, niezbyt droga technologia, zapewniająca lepsze zabezpieczenie przed korozją niż pomalowanie. Możesz ją dostrzec na co dzień: prawie wszystkie kolanka, złączki, a także rury instalacji gazowych wykonuje się ze stali ocynkowanej.

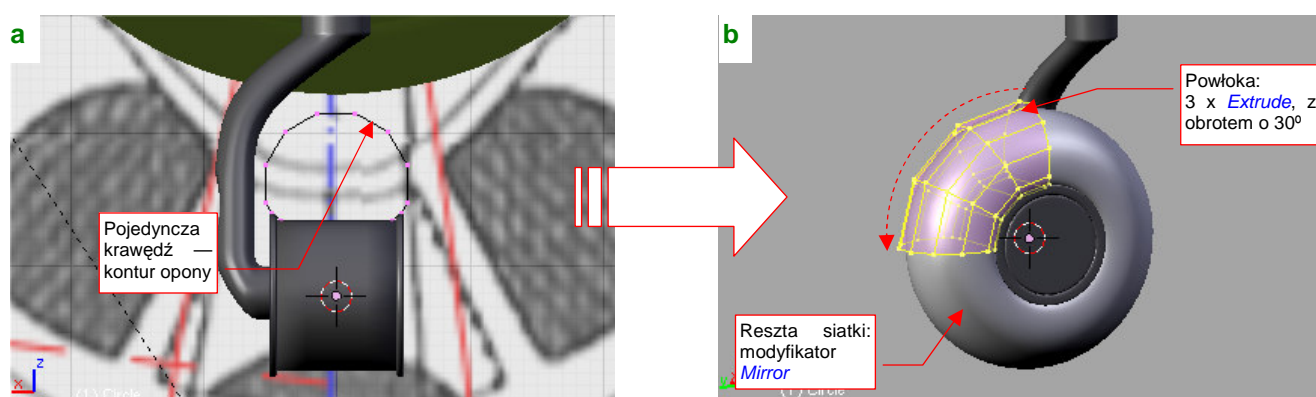
Piastę koła najszybciej jest uformować, zaczynając od jej przekroju poprzecznego (Rysunek 7.1.6a):



Rysunek 7.1.6 Formowanie piasty kółka ogonowego

Potem wystarczy wytłoczyć ten kontur poprzez kolejne obroty o  $30^\circ$  wokół lokalnej osi **X** obiektu (Rysunek 7.1.6b). Na koniec dodałem do siatki modyfikator *Mirror*, wzdłuż osi **Y** i **Z**. Piastę nazwałem **B.515.Gear.Tail** i przypisałem jej ten sam materiał, co goleni (**B.Steel.Rough**).

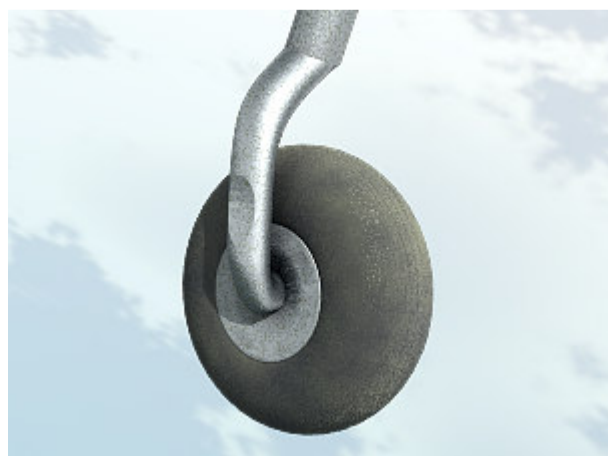
Tak samo uformowałem oponę (**B.520.Tyre.Tail**). Zacząłem od linii konturu (Rysunek 7.1.7a), a potem wytłoczyłem ją w „ćwiartkę” koła (Rysunek 7.1.7b):



Rysunek 7.1.7 Formowanie opony kółka ogonowego

Oponie kółka ogonowego przypisałem właściwości „zakurzonej gumy” — materiał **B.Rubber.Tyre**. Gdybyś miał wątpliwości, jak powinna wyglądać, po prostu przyjrzyj się uważnie kołom jakiegoś niezbyt nowego samochodu.

Rysunek 7.1.8 przedstawia próbny rendering tego zespołu. Muszę przyznać, że uzyskanie właściwego efektu zużycia koła nie było łatwe. Wymagało wykorzystania wielu tekstur proceduralnych. Najpierw ustaliłem bazowe własności materiału opony, takie jak barwa, rozpraszanie światła (tylko zupełnie nowa guma jest czarna!) i połyskliwość. Potem za pomocą dwóch tekstur proceduralnych naniosłem efekty chropowatości i nieznacznego zakurzenia. Następnie nałożyłem maskę, która „przepuszcza” kolejne tekstury tylko wzdłuż bieżnika. Jako, że bieżnik jest zawsze bardziej wytarty, powtórzyłem na nim, dla wzmocnienia, efekty chropowatości i zabrudzeń. Dokładny opis, jak przygotować taki materiał dla opony, znajdziesz na str. 998.

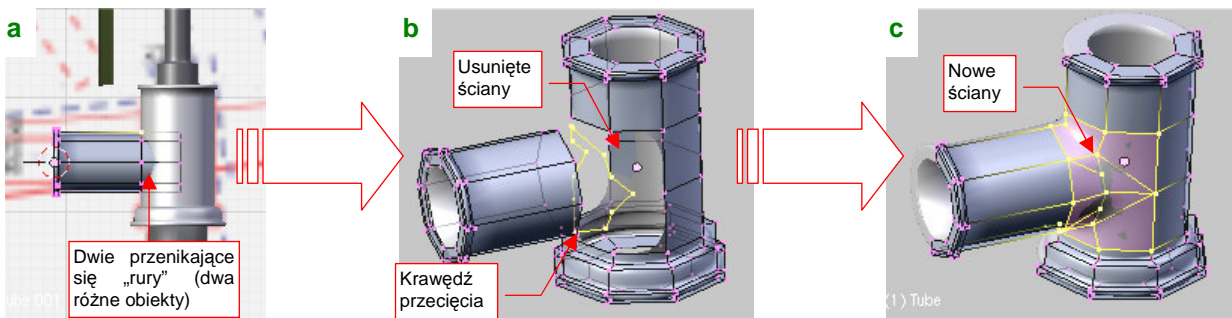


Rysunek 7.1.8 Próbny rendering z materiałem **B.Rubber.Tyre**

Mówiąc szczerze, zawsze zaczynam modelowanie takich mechanizmów, jak podwozie, od odrębnego szkicu. To taka moja metoda na dogłębne poznanie rzeczy, którą mam odtworzyć. Konieczność narysowania poszczególnych detali (nawet niestarannie!) zmusza mnie do zastanowienia się nad ich kształtem i rolą. W czasie szkicowania decyduję, na jakie części podzielę cały zespół, oraz jakie szczegóły pominę. Po zakończeniu pracy taki szkic jest już niepotrzebny i ląduje w koszu. Przypuszczam, że możesz spróbować innych metod „rozumienia” modelowanego mechanizmu. Najważniejsze w każdym razie, by najpierw ułożyć sobie w głowie plan pracy, a potem go zrealizować. Nie zaczynaj takich rzeczy impulsywnie, nie kieruj się naszym tradycyjnym (narodowym?) podejściem „jakoś to będzie” — bo zmarnujesz niepotrzebnie swój czas.

Po bliższej analizie stwierdzisz zapewne, że nawet najprostsze podwozie samolotu składa się z kilkunastu części, o różnym kształcie. Oznacza to, że trzeba tworzyć je co najmniej w tempie kilku dniennie, aby całość prac nie rozciągnęła się w tygodnie. Nie zawsze jest to takie łatwe. Muszę przyznać, że wymyślenie w miarę prostej siatki dla każdego niebanalnego detalu zajmuje mi trochę czasu. Na tej i następnych stronach pokażę formowanie szczegółów dwóch części: wahacza i amortyzatora. W ten sposób będziesz mógł skorzystać z moich doświadczeń, i podpatrzeć typowe metody tworzenia takich detali. Resztę elementów zespołu kółka ogonowego omówię bardzo pobieżnie — po jednej ilustracji na każdy.

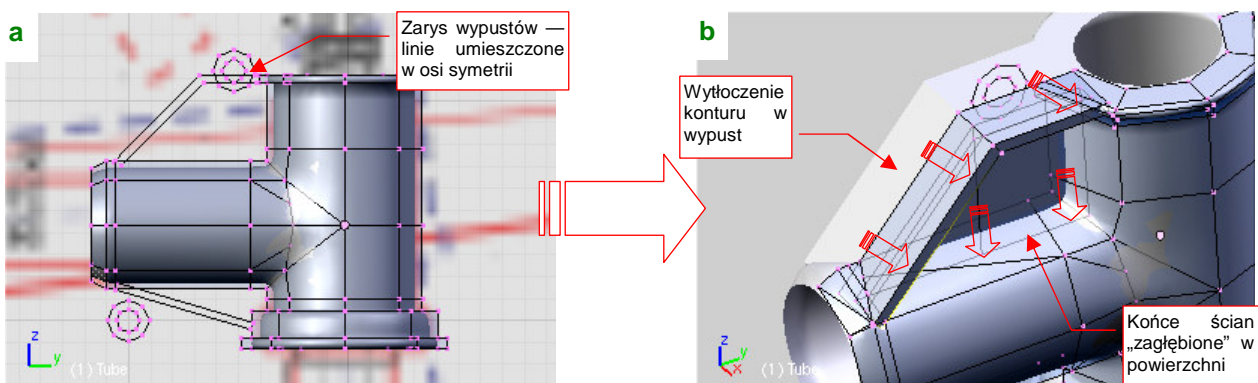
Zacznijmy od końcówki wahacza, w której ma być osadzona goleń podwozia (Rysunek 7.1.9):



Rysunek 7.1.9 Początek formowania wahacza — miejsce na goleń podwozia

Podstawowy kształt końcówki to dwie rury, połączone w poziome „T” (Rysunek 7.1.9a). Każda z nich jest początkowo oddzielnym obiektem, bym mógł wyznaczyć ich krawędź przecięcia (za pomocą skryptu [Cross Section](#) — por. str. 803). Po wyznaczeniu tej krawędzi scalałem (**Ctrl-J**) obydwie siatki, i usunąłem ściany otaczające linię przenikania połówek (Rysunek 7.1.9b). Utworzyłem w ich miejsce nowy układ ścian, włączając w siatkę krawędź przecięcia (Rysunek 7.1.9c). Przy okazji usunąłem jedną z połówek siatki i włączyłem modyfikator [Mirror](#).

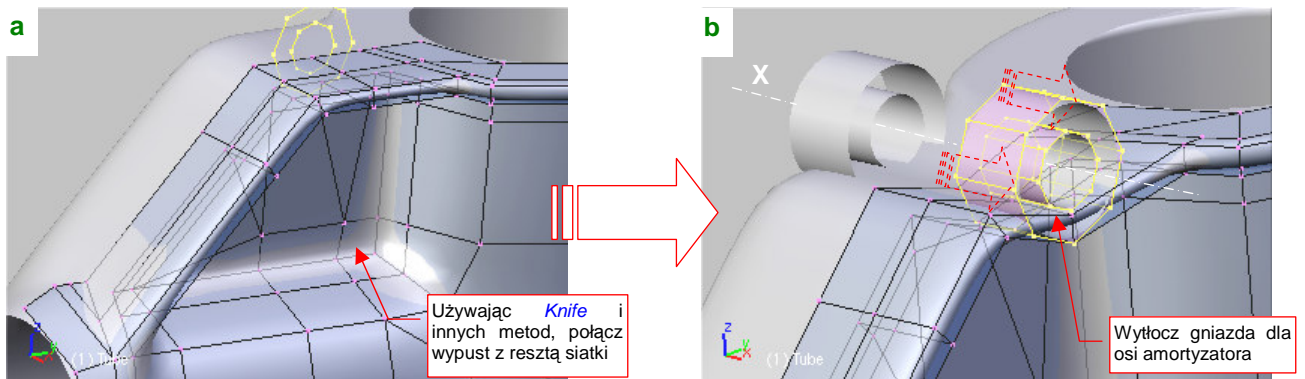
Kolejnym utrudnieniem są wypusty, umieszczone u góry i dołu końcówki. Zacząłem ich formowanie od zarysu w rzucie z boku. Do odwzorowania okręgów użyłem ośmioboków (Rysunek 7.1.10a):



Rysunek 7.1.10 Wahacz — początek formowania wypustu

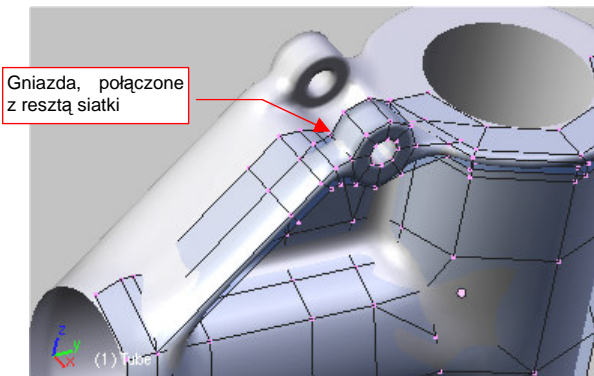
Potem wytłoczyłem kontury wypustów na boki i do dołu, by uformować ich zasadniczy kształt (Rysunek 7.1.10b). Zrobiłem to tak, by wytłoczone ściany zagłębły się wszędzie w powierzchni „rur” końcówki.

Kolejnym krokiem jest scalenie siatki „nur” i wypustu. Zrobiłem to, tworząc nacięcia *Knife*, a potem zmieniając układ niektórych krawędzi i ścian (Rysunek 7.1.11a):

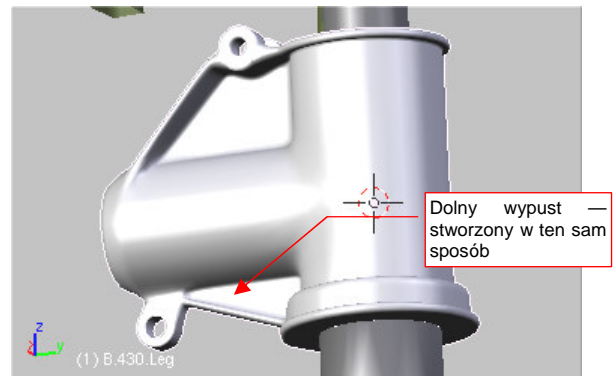


Rysunek 7.1.11 Wahacz — formowanie wypustu

Następnie przesunąłem w odpowiednie miejsce zarys gniazda osi amortyzatora, i wytłoczyłem je wzdłuż osi **X** (Rysunek 7.1.11b). Tymi samymi metodami, których użyłem poprzednio do scalenia wypustu z resztą siatki, scalałem teraz z nią cylindry gniazd osi (Rysunek 7.1.12):



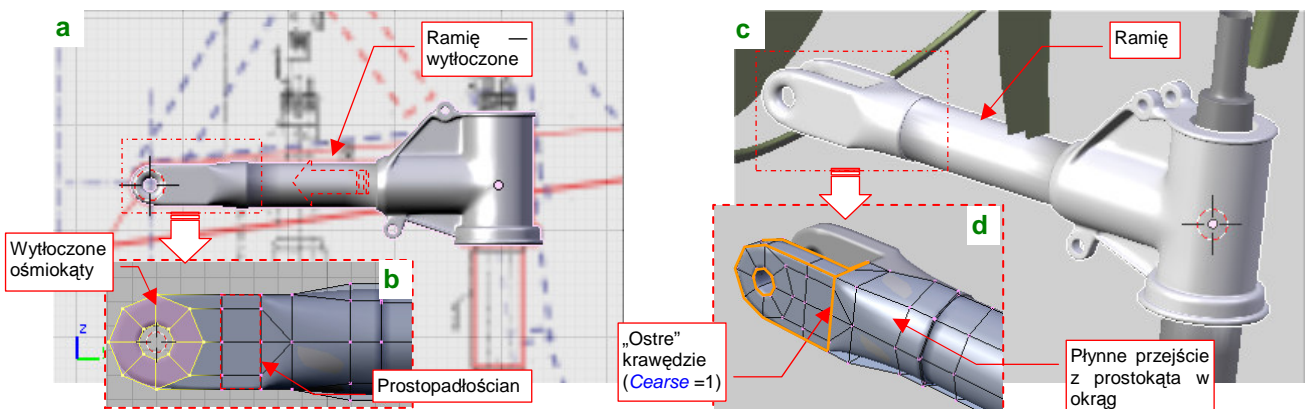
Rysunek 7.1.12 Gotowy wypust górny



Rysunek 7.1.13 Wahacz — gotowa końcówka

I to wszystko — górny wypust jest gotowy. W ten sam sposób uformuj drugi, u dołu wahacza (Rysunek 7.1.13).

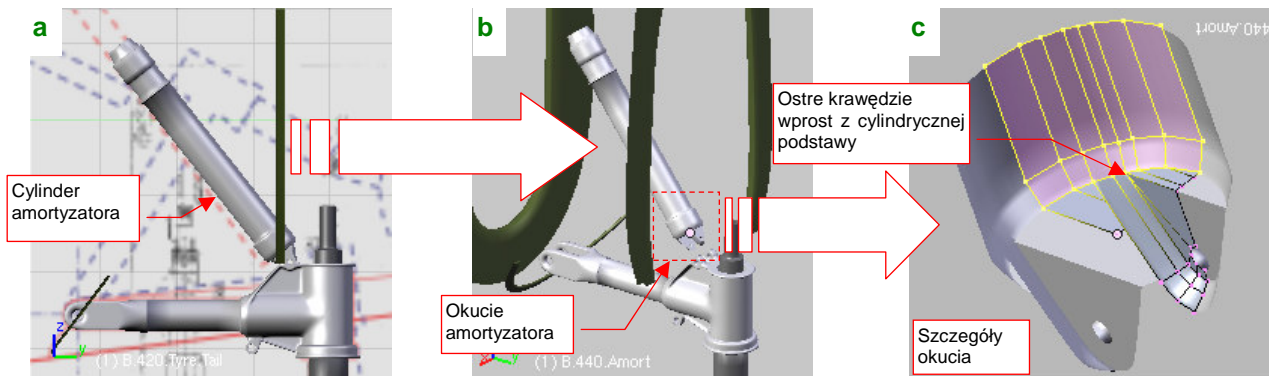
Do wykonania pozostała jeszcze reszta wahacza: ramię, które wytłoczę z końcówki, oraz okucie osi obrotu (Rysunek 7.1.14a):



Rysunek 7.1.14 Wahacz — okucie osi obrotu

Okucie osi obrotu wahacza wytłoczyłem z dwóch współosiowych ośmiokątów (Rysunek 7.1.14b). Potem połączyłem je z prostopadłościanem, który z kolei płynnie łączy się z okrągłym ramieniem wahacza (Rysunek 7.1.14d). Takie końcówki napotkasz w wielu elementach tego modelu. Temu wahaczowi nadałem nazwę **B.530.SwingArm**.

Amortyzator wydaje się na pierwszy rzut oka walcem, zupełnie łatwym do uformowania (Rysunek 7.1.15a). Ot, wystarczy stworzyć cylinder, rozciągnąć, dodać kilka wytłoczeń w odpowiednich miejscach, a na koniec pochylić, i gotowe. A jednak, gdy przyjrzyś się bliżej jego dolnej (tej lepiej widocznej z zewnątrz!) części (Rysunek 7.1.15b), natokasz tam niby proste okucie, którego wykonanie może przysporzyć niespodziewanych kłopotów (Rysunek 7.1.15c):

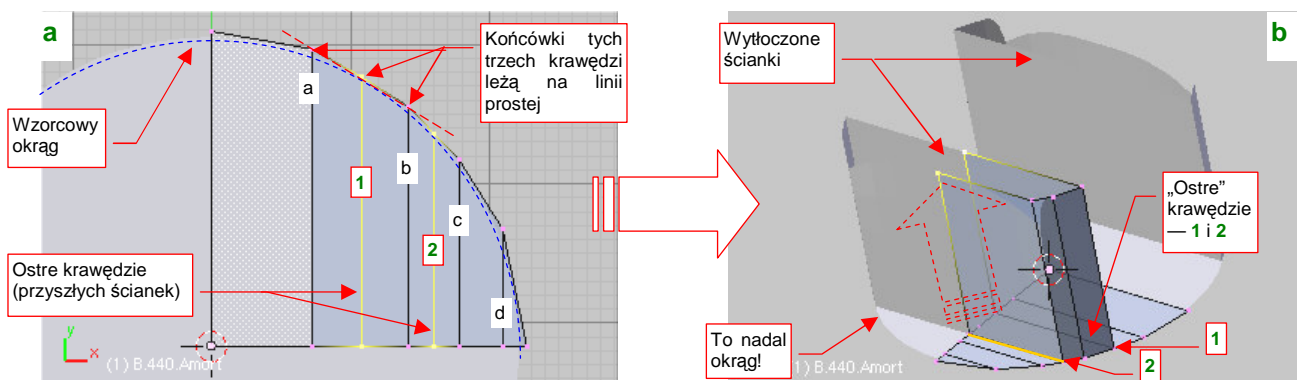


Rysunek 7.1.15 Amortyzator i jego okucie

Problem z kształtem tego okucia polega na tym, że jego ścianki „wyrastają” wprost z okrągłej podstawy (Rysunek 7.1.15c). Takie ostre załamania psują zawsze kształt wygładzonych powierzchni podziałowych. Jak więc uzyskać w tym miejscu taki „kant”, a jednocześnie zachować wygładzony, okrągły kształt krawędzi?

Można to zrobić całkiem łatwo, pod warunkiem że zaczniemy formowanie całego amortyzatora właśnie od tego okucia. Wstaw do rysunku jakiś okrąg, który będzie nam służył za wzorec kształtu. Potem wstaw drugi okrąg (np. ośmiokąt) — to początek amortyzatora. Możesz od razu dodać do siatki odpowiednie modyfikatory *Mirror* w kierunkach **X**, **Y**, i usunąć niepotrzebne wierzchołki, abyśmy dalej pracowali na „ćwiartce” okręgu. (Im mniej punktów do zmiany, tym lepiej!). Wytłocz (*Extrude* → *Edges*) krawędź ośmiokąta, zmieniając jej skalę wzdłuż osi **Y** do zera, by wypełnić „ćwiartkę” koła (Rysunek 7.1.16a).

Następnie wstaw w naszą „ćwiartkę” dodatkowe krawędzie przyszłych ścianek okucia: **1** i **2**. Obok nich dodaj kolejne, pomocnicze krawędzie: **a** i **c** (**b** już istniała — to był wierzchołek oryginalnego ośmiokąta) :



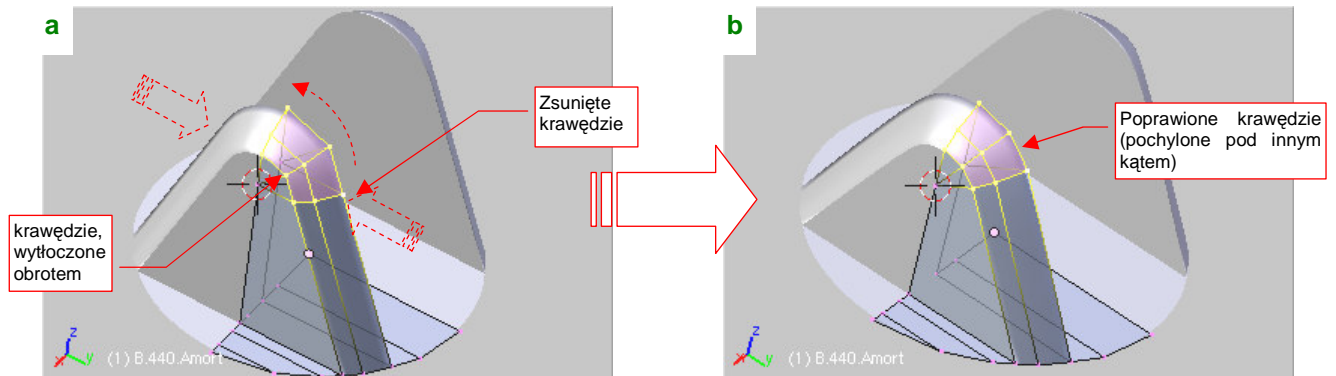
Rysunek 7.1.16 Końcówka amortyzatora — początki formowania

Jest taka własność powierzchni podziałowych: jeżeli końcówki „ostrej” krawędzi (np. **1**) i jej „sąsiadów” (**a**, **b**) leżą na linii prostej — to powłoka pozostanie w tym miejscu gładka (pokazuje to Rysunek 7.1.16a). Wystarczy teraz poprzesuwać trochę wierzchołki, by dopasować kształt obrysu do wzorcowego okręgu. Aby jak najlepiej odwzorować łuk, dodaj do siatki ostatnią, pomocniczą krawędź **d**. Całe dopasowanie zajmuje tylko chwilę. Potem możesz zaznaczyć krawędzie **1** i **2** jako „ostre” (*Cease* = 1). Sam zobaczysz, że nie zaburzy to w żaden sposób kształtu okręgu obwodu.

Teraz spokojnie możesz już wytoczyć (*Extrude* → *Region*) powierzchnię ograniczoną krawędziami **1** i **2** w ścianki okucia (Rysunek 7.1.16b).



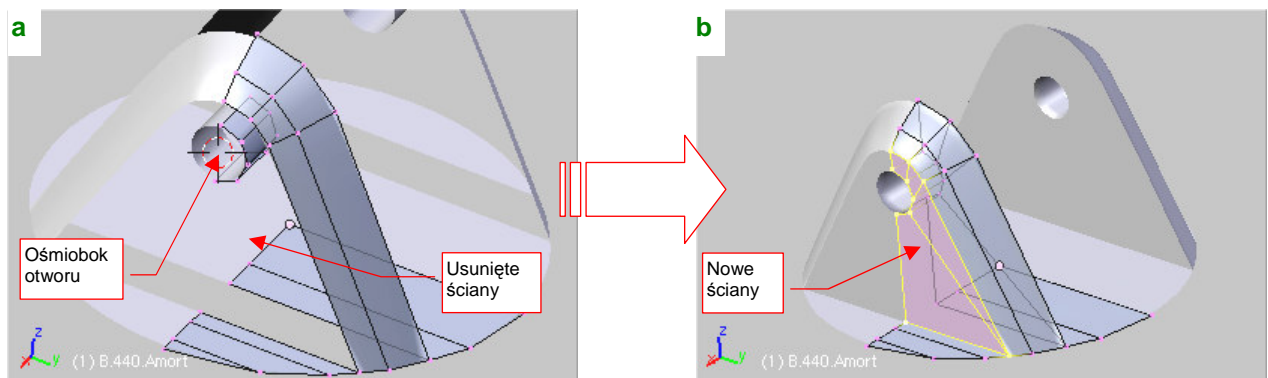
Końcówki wytłoczonych ścianek „zsuń” wzdłuż osi **Y** do środka, by nadać im kształt trójkąta. Następnie ustaw kursor 3D w wierzchołku leżącym w na płaszczyźnie symetrii — tu będzie biegła oś otworu. Wytłocz krawędź końcówek w dwóch obrotach w półokrągłe zakończenie (Rysunek 7.1.17a):



Rysunek 7.1.17 Końcówka amortyzatora — formowanie wypustu

Potem obróć nieco te krawędzie końcówki jeszcze raz, aby dopasować je lepiej do pochylenia ścianek okucia (Rysunek 7.1.17b).

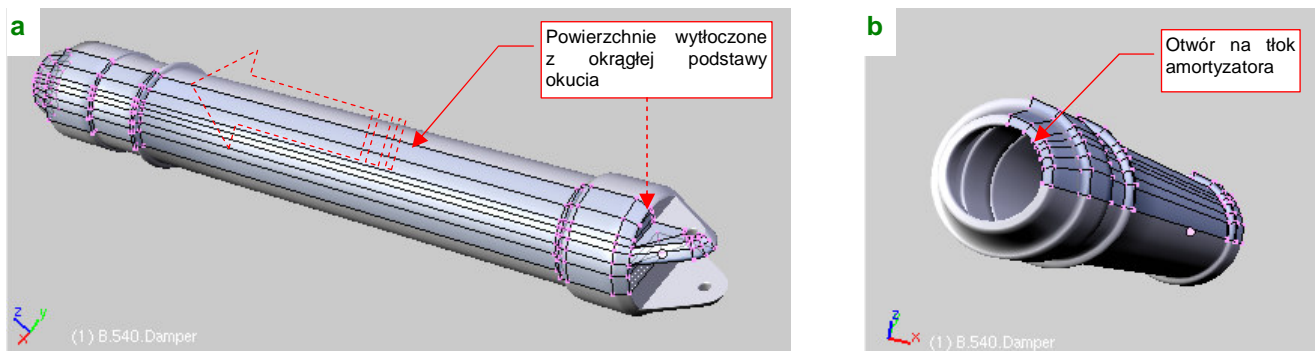
Następnie usuń dwa wierzchołki z centrum ścianki, by zniknęły dotychczasowe ściany boczne, i wstaw w ich miejsce kolejny, ośmioboczny cylinder (Rysunek 7.1.18a):



Rysunek 7.1.18 Końcówka amortyzatora — formowanie wypustu

Połącz krawędzie ścian otworu ze zewnętrznymi ścianami siatki, uzyskując w ten sposób nowe powierzchnie boczne (Rysunek 7.1.18b). Potem pozostaje tylko zaznaczyć je jako ostre, oraz zmienić tryb cieniowania na **Solid**, by okucie było gotowe.

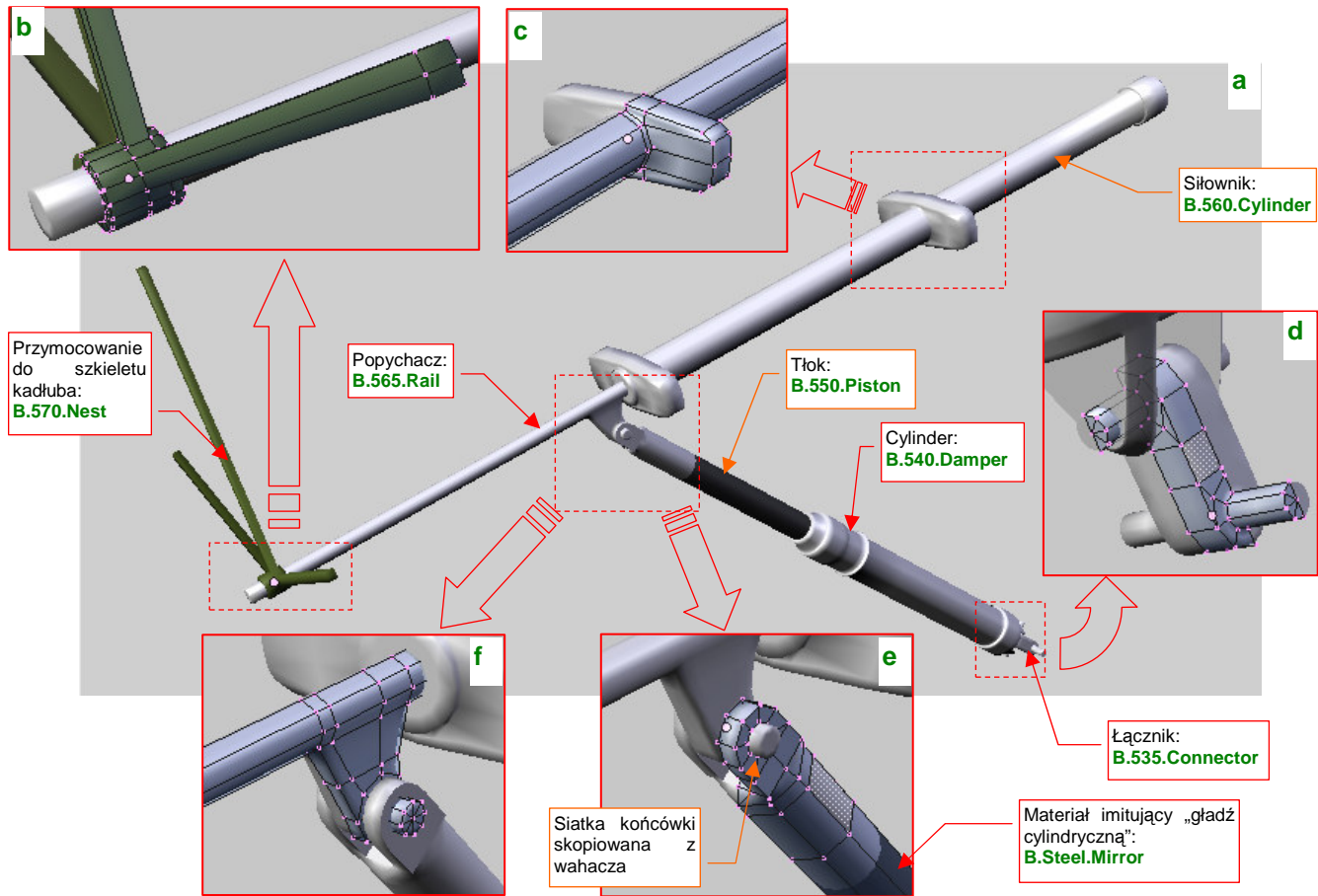
Resztę siatki amortyzatora uzyskasz przez wytłoczenie okrągłej krawędzi podstawy okucia (Rysunek 7.1.19a):



Rysunek 7.1.19 Gotowy cylinder amortyzatora (B.540.Damper)

Na drugim końcu tego cylindra pozostaw otwór na tłok (Rysunek 7.1.19b). Nadaj tej części nazwę „z serii 500”, np. **B.540.Damper**.

Rysunek 7.1.20 przedstawia elementy podzespołu amortyzatora (por. str. 421, Rysunek 7.1.1):

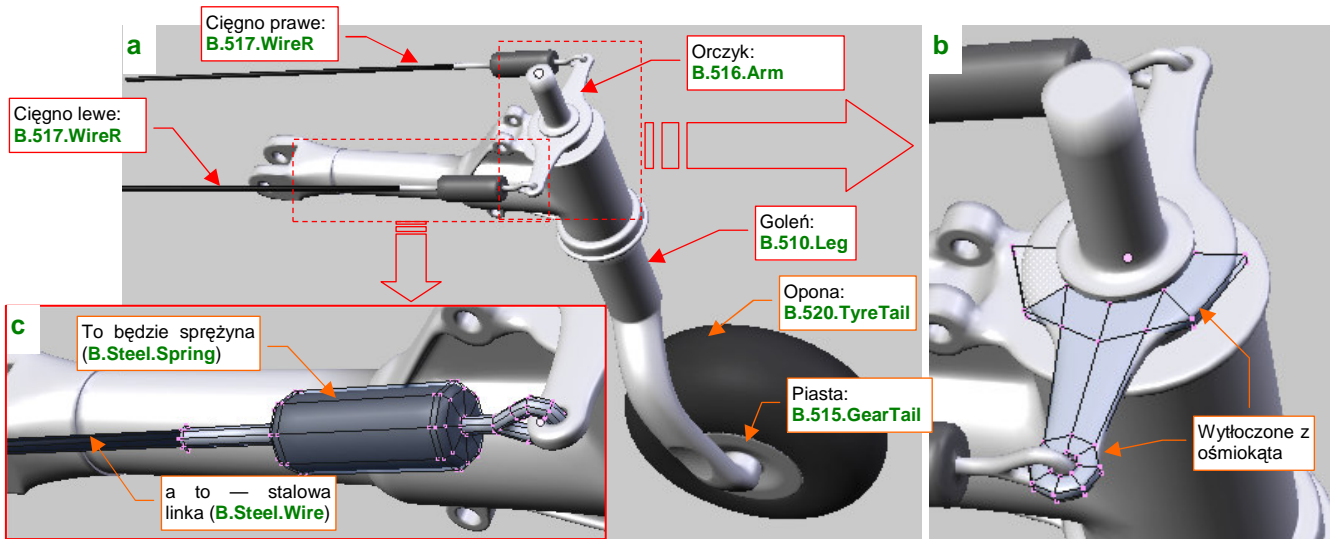


Rysunek 7.1.20 Podzespół amortyzatora

Cylinder amortyzatora był połączony z wahaczem poprzez dodatkowy łącznik (**B.535.Connector** — Rysunek 7.1.20d). Taki element umożliwiał dodatkowo zginanie się cylindra na boki. (Przypuszczam, że czasami wahaczem mogło całkiem porządnie „machnąć” w poprzek kadłuba, szczególnie na jakimś nierównym lądowisku). Siatka **B.335.Connector** powstała z połączenia dwóch wytłoczonych w cylindry ośmiokątów. Element jest niewielki, więc jest tylko wygładzona „zgrubnym” modyfikatorem *Subsurf* (*Level* = 1). Aby oszczędzić sobie pracy, zamodelowałem tylko „ćwiartkę” tej części, a resztę załatwiły za mnie modyfikatory *Mirror* wzdłuż lokalnych osi **X** i **Z** (Rysunek 7.1.20d). Wydaje mi się, że takie elementy były pozostawione w kolorze „surowej” stali — więc zastosowałem tu gładzki materiał **B.Steel.Smooth** (to **B.Steel.Rough** bez tekstury nierówności — p. str. 992). Tłok amortyzatora (**B.550.Piston**) to prosty cylinder, zakończony „widelcem” z otworem na oś (Rysunek 7.1.20e). Siatkę widelca skopiowałem (**P** — *Separate*) z wykonanej wcześniej końcówki wahacza. Potem wystarczyło ją odpowiednio ustawić, zmniejszyć, i złączyć (**Ctrl-J** — *Join Meshes*) z siatką cylindra. Zwróć uwagę, że obszar roboczy cylindra ma bardzo gładką powierzchnię. Odzworowałem to, stosując w tym miejscu inny materiał: **B.Steel.Mirror** (to taka trochę ciemniejsza odmiana **B.Steel.Smooth**, bez tekstury zabrudzeń). Siłownik (**B.560.Cylinder**) to walec, z którego w miejscu mocowań wytłoczyłem poprzeczne „skrzydełka” (Rysunek 7.1.20c). Grubość tych wypustów jest dopasowana do szerokości siatek żeber — tak, aby razem wyglądały jak zupełnie cienkie, blaszane wzmocnienia. Popychacz siłownika (**B.565.Rail**) to także cylinder, z którego w odpowiednim miejscu wytłoczyłem wypust z osią (Rysunek 7.1.20f). Przyłączyłem do niej cylinder amortyzatora. Ostatnim, najmniej widocznym z zewnątrz elementem całego zespołu, są wysięgniki (**B.570.Nest**). Mocują koniec popychacza do szkieletu kadłuba. W odróżnieniu od pozostałych części nie zastosowałem tu żadnego wygładzenia — bo i tak ten element będzie tylko majaczył w głębi kadłuba.

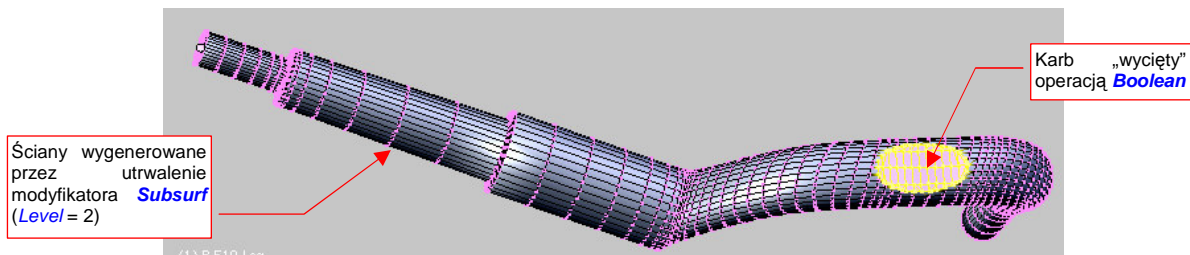
**B.535.Connector** jest „rodzicem” (*Parent*) cylindra amortyzatora (**B.540.Damper**). Siłownik (**B.560.Cylinder**) jest „rodzicem” popychacza (**B.565.Rail**), a ten z kolei — „rodzicem” tłoka amortyzatora (**B.550.Piston**).

Rysunek 7.1.21a) przedstawia zespół wahacza, wraz z cięgnami służącymi do skręcania kółka ogonowego (były sprzężone ze sterem kierunku):



Rysunek 7.1.21 Podzespół wahacza

Siatki wahacza (**B.530.SwingArm**), opony (**B.520.TyreTail**), piasty (**B.515.GearTail**) i goleni (**B.510.Leg**) prezentowałem już w tej sekcji, więc nie ma co ich tu drugi raz pokazywać. Chociaż co do goleni, to muszę odnotować, że zdecydowałem się już ostatecznie „utrwalić” jej karb, uzyskany za pomocą modyfikatora *Boolean* (por. str. 422 i 909). Dlaczego teraz? Bo wydaje mi się, że nie będzie już trzeba poprawiać jej kształtu<sup>1</sup>. Aby utrwalić modyfikator *Boolean*, trzeba utrwalić także *Subsurf*. To dodaje goleni tyle wierzchołków, że można się pożegnać z myślą o jakichkolwiek dalszych poprawkach tej siatki (Rysunek 7.1.22):



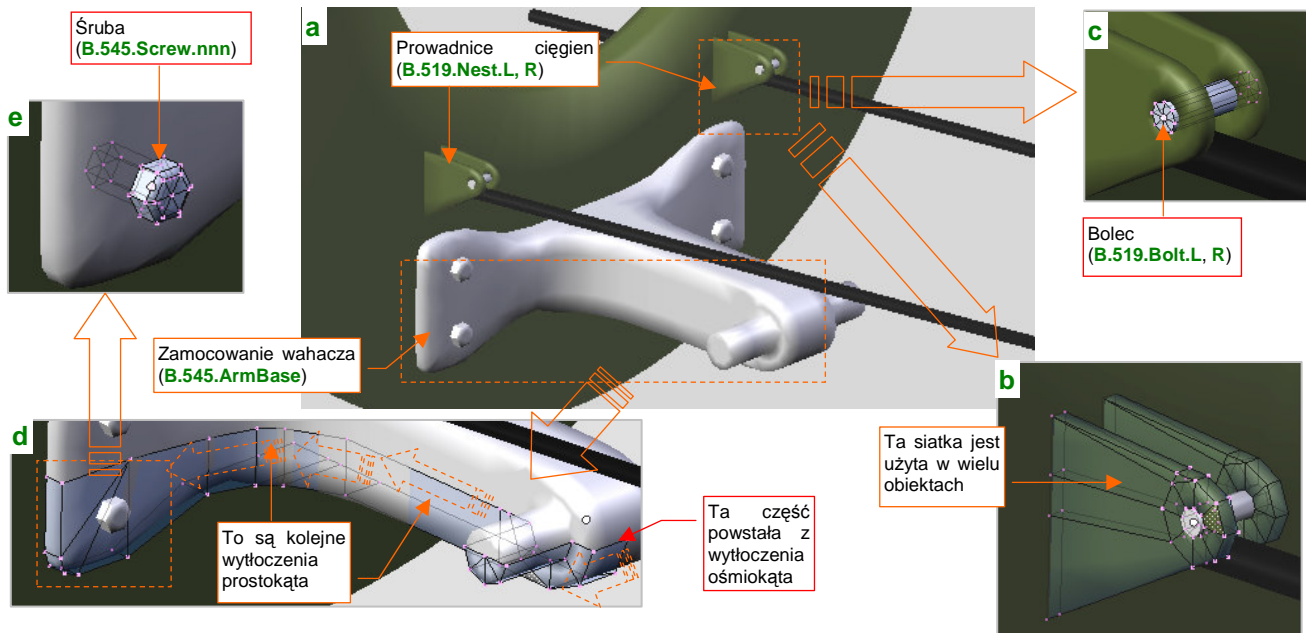
Rysunek 7.1.22 Siatka goleni (**B.510.Leg**) po „utrwaleniu” modyfikatorów *Subsurf* i *Boolean*

Elementem zespołu pokazywanego przez Rysunek 7.1.21a), którego do tej pory jeszcze nie omawialiśmy, jest orczyk (**B.516.Arm**), przymocowany do końca goleni. Wytłoczyłem go z dwóch ośmiokątów, które następnie połączyłem (Rysunek 7.1.21b). Zastosowałem tu symetrię wzdłuż osi **X** i **Z** (można było i wzdłuż **Y**, ale nim o tym pomyślałem, miałem już element gotowy). Podobnie jak łącznik wahacza z amortyzatorem (**B.535.Connector**), orczyk jest wykonany z „surowej” stali (materiał **B.Steel.Smooth**). Cięgna — lewe i prawe (**B.517.Wire.L** i **B.517.Wire.R**) używają tej samej siatki **B.517.Wire**. Sama siatka to odpowiednio wytłoczone cylindry (Rysunek 7.1.21c). „Rurki”, modelujące cięgna, mają przypisany specjalny materiał **B.Steel.Wire**. Na renderze wyglądają dzięki niemu jak plecione linki stalowe (szczegóły — str. 993) Zwróć uwagę na zgrubienia przy orczyku — za pomocą odpowiedniego materiału (**B.Steel.Spring**) zmienię je na renderingu w realistyczne sprężyny (szczegóły — p. str. 994).

Jeżeli chodzi o hierarchię: wahacz (**B.530.SwingArm**) jest „rodzicem” goleni (**B.510.Leg**). Goleń jest z kolei „rodzicem” piasty koła (**B.515.GearTail**) i orczyka (**B.516.Arm**). Piasta jest „rodzicem” opony (**B.520.TyreTail**), a orczyk — cięgien (**B.517.Wire.R** i **L**).

<sup>1</sup> Gdyby jednak trzeba było je jeszcze poprawić, skopiowałem (**Shift-D**) oryginalne: goleń z włączonymi modyfikatorami i obiekt „tnącym”, i przeniósłem te kopie (*Object→Make Links→To Scene...*) do sceny *Spares*. To nasz schowek na różne elementy, które mogą się jeszcze przydać.

Jako kolejny zespół można wyróżnić elementy, przymocowane „na sztywno” do wzmocnionej wręgi kadłuba (Rysunek 7.1.23a) :

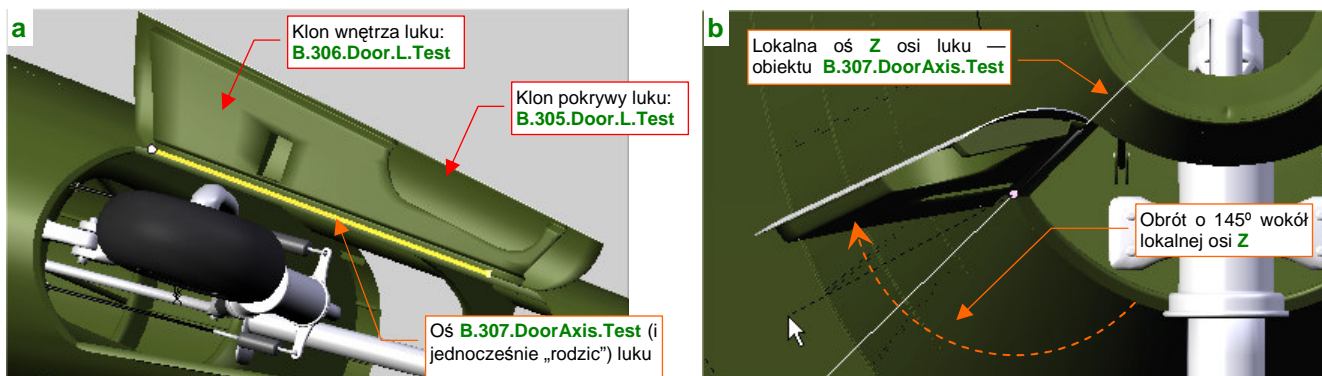


Rysunek 7.1.23 Elementy przymocowane do wręgi ogonowej

Najważniejszą częścią jest tu okucie **B.545.ArmBase**, mocujące wahacz do wręgi. Oś obrotu wahacza wytłoczyłem z ośmiokąta, a resztę — za pomocą kolejnych wytłoczeń sekcji prostokątnej (Rysunek 7.1.23d). Formowałem tylko symetryczną „ćwiartkę” okucia, a potem ją powieliłem za pomocą modyfikatora *Mirror* wzdłuż lokalnych osi **X** i **Z**. W tym miejscu były bardzo widoczne łby czterech śrub, więc nie pozostało mi nic innego, jak je zamodelować (**B.545.Screw.001 - .004**) . Śruby przydadzą się w dziesiątkach innych miejsc, więc utworzyłem je jako cztery klony tej samej siatki — **B.Screw**. Sama siatka jest bardzo prosta (Rysunek 7.1.23e). Zwróć uwagę na położenie punktu centralnego obiektu — jest w płaszczyźnie podstawy łba śruby. Tak jest najwygodniej. „Rodzicem” wszystkich czterech śrub jest oczywiście okucie **B.545.ArmBase**.

Kolejnymi składnikami zespołu pokazywane przez Rysunek 7.1.23a są prowadnice cięgien (**B.517.Nest.L** i **.R**). Takie trójkąty z otworem na końcu przydadzą się jeszcze w innych miejscach, więc, podobnie jak śruby, są to klony jednego wzorca (Rysunek 7.1.23b). Łączy je bolec (**B.519.Bolt.L, .R**) . Siatka tego cylindra także będzie zawsze powieleniem jednego pierwowzoru: **B.Bolt**. Jeżeli chodzi o hierarchię, to bolec (**B.519.Bolt.L** lub **.R**) i prawa ścianka (**B.519.Nest.002.L** lub **.R**) są przypisane do lewych ścianek (**B.519.Nest.001.L** i **.R**), a te — do wręgi kadłuba.

Nim zabierzemy się za formowanie popychaczy luków podwozia, sklonujmy (**Alt-D**) do testów jeden luk wraz z osią (**B.307.DoorAxis.R**) (Rysunek 7.1.24a):

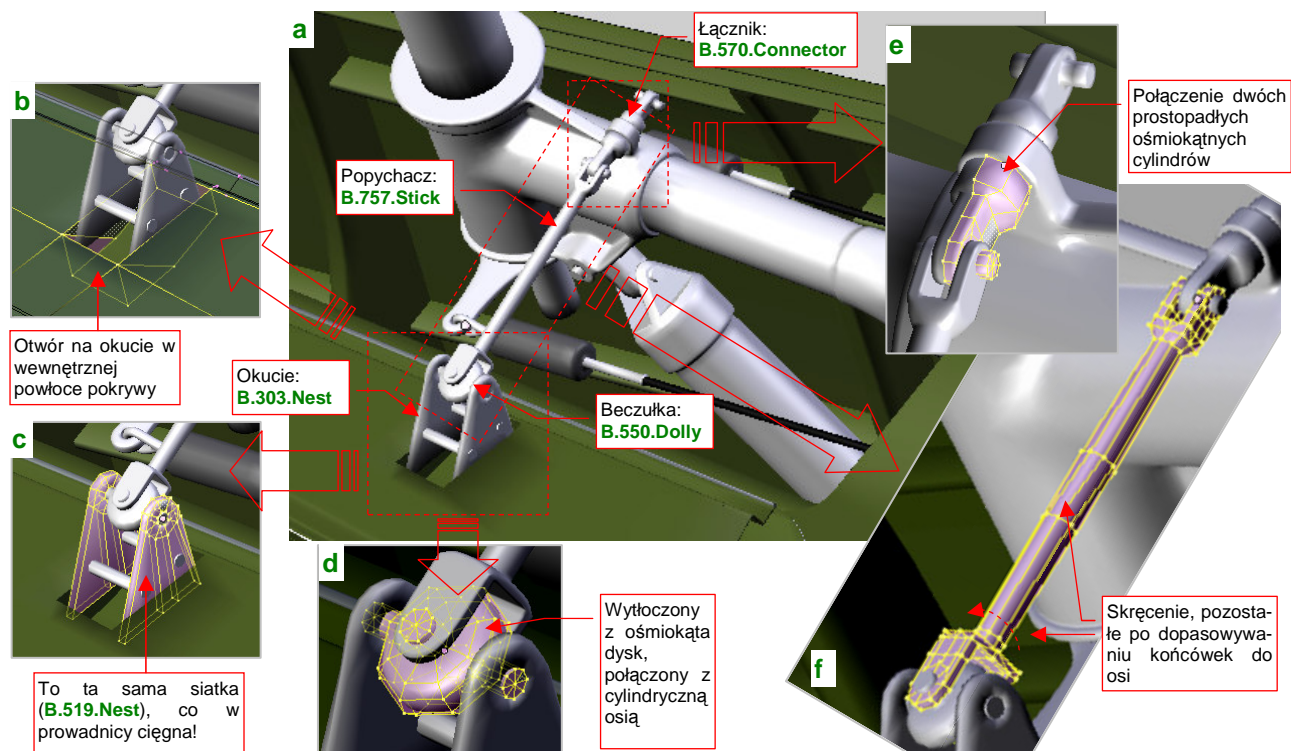


Rysunek 7.1.24 Robocza kopia luku podwozia i jego osi

Luk po skopiowaniu otwórz, obracając oś **B.307.DoorAxis** o jakiś „równy” kąt — np. 145° (Rysunek 7.1.24b).

Jeżeli obrócisz zespół luku podwozia o jakiś przypadkowy kąt, będziesz musiał potem go z powrotem pracownice dopasowywać do pozycji „zamkniętej”. Dlatego — na wszelki wypadek — lepiej jest stworzyć robocze klony pokryw i do nich przymierzać otwarte podwozie. Ustaw taki klon luku w pozycji „otwarte” i dopasuj go do kółka ogonowego. Przekonasz się, że to bardzo wygodne. Możesz np. zmieniać kształt jego siatki, a zmiany są od razu widoczne na pierwowzorach pokryw. Konieczność takiej zmiany przytrafiła się i mi: podczas formowania zapomniałem wykonać w wewnętrznej powierzchni pokrywy podwozia otworu na okucie popychacza (Rysunek 7.1.25b). Musiałem ją uzupełnić teraz, oczywiście przymierzając od razu okucie do właściwych rozmiarów.

Rysunek 7.1.25a) pokazuje szczegóły popychacza pokryw kółka ogonowego:



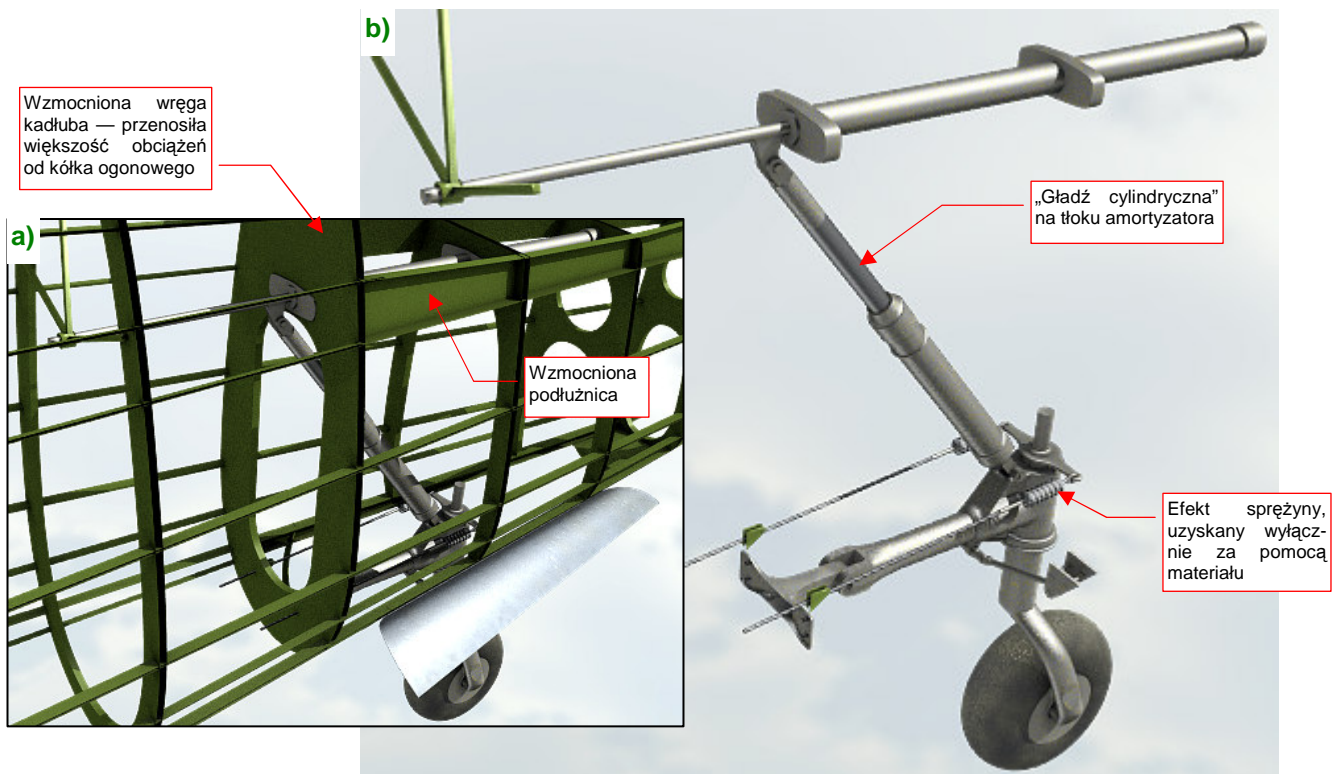
Rysunek 7.1.25 Zespół popychaczy luku podwozia

Popychacze lewy i prawy (**B.757.Stick.L** i **.R**) są połączone ze sobą poprzez otwór w wahaczu łącznikiem **B.570.Connector** (Rysunek 7.1.25e). Ten łącznik może się obracać — bo inaczej popychacze by się zablokowały podczas chowania podwozia. Każda z końcówek popychacza **B.575.Stick** ma inny kształt i jest pochylona pod innym kątem, by dopasować się do „swojej” osi. Uzyskałem to poprzez obrót części siatki wokół lokalnej osi **Z** obiektu. Śladem p tych zabiegach jest skrzyżowana siatka pośrodku popychacza (Rysunek 7.1.25f).

Okucie pokrywy podwozia zbudowałem z klonów tej samej trójkątnej siatki „blaszki” (**B.519.Nest**), co przewodnice ciągną (por. str. 430, Rysunek 7.1.23b). Skoro wszystkie te obiekty używają tej samej siatki, to jak udało mi się przypisać im inny kolor? To nic trudnego: na okuciu pokryw przełączyłem tryb przypisania materiału z **ME**(sh) na **OB**(ject) (szczegóły — str. 959). W trybie **OB** renderer używa materiału przypisanego do obiektu, a nie siatki. Dzięki temu każdy z klonów może mieć inny kolor, bo jest odrębnym obiektem. Do każdego z okuc **B.303.Nest.L** i **.R** dodałem u dołu dwa dodatkowe bolce (**B.303.Bolt.L.nnn** lub **.R.nnn**) — bo są widoczne na zdjęciach. W osi okucia umieściłem „beczułkę” (**B.550.Dolly**). Ten element umożliwia obrót końcówki popychacza w dwóch płaszczyznach. Mówiąc szczerze, nie jestem pewien kształtu tego elementu — na wszystkich fotografiach, które posiadam, jest zasłonięty przez blaszki okucia.

Hierarchia tego zespołu wygląda następująco: centralny łącznik (**B.570.Connector**) jest „rodzicem” popychacza (**B.575.Stick**). („Rodzicem” centralnego łącznika jest wahacz — **B.530.SwingArm**). Wewnętrzna część pokrywy luku podwozia (**B.306.Door**) jest „rodzicem” okucia (**B.303.Nest**). Okucie jest „rodzicem” beczułki (**B.550.Dolly**) oraz bolców wzmacniających.

Rysunek 7.1.26a) przedstawia uformowany mechanizm kółka ogonowego, wkomponowany w szkielet kadłuba, a Rysunek 7.1.26b) — bez szkieletu (widać wtedy więcej szczegółów):



Rysunek 7.1.26 Mechanizm kółka ogonowego — „w szkielecie” i osobno

Teraz konstrukcja ogona kadłuba stała się bardziej zrozumiała. Wręga przed lukiem podwozia musiała być taka szeroka (wzmocniona), bo przenosiła większość obciążeń pochodzących od kółka ogonowego. Skrzynkowe, wzmocnione podłużnice pod statecznikiem poziomym tworzyły wraz z tą wręgą „ramę”, w której zamocowany był siłownik.

Rysunek 7.1.27 przedstawia, jak teraz wygląda wnętrze luku podwozia (zewnątrzna powierzchnia kadłuba jest niewidoczna). Przyznasz, że prezentuje się już bardziej realistycznie, prawda?



Rysunek 7.1.27 Widok do wnętrza komory kółka ogonowego

Myślę, że nie będziemy już dodawać tutaj dalszych detali. Oczywiście, w następnej sekcji uzupełnimy jeszcze tylko brakujący, prawy popychacz pokryw, wraz z okuciem.

Uff, to była bardzo długa sekcja. Staralem się w niej pokazać ze wszystkimi szczegółami, jak modeluje się podwozie samolotu. Wybrałem do tego przykładu stosunkowo prosty zespół kółka ogonowego. Gdy w dalszych sekcjach będę omawiał podwozie główne, pomnę już szczegóły formowania każdej śrubki czy popychacza. Po zapoznaniu się z poprzednimi stronami, powinieneś być w stanie tworzyć je szybko, bez żadnych podpowiedzi.

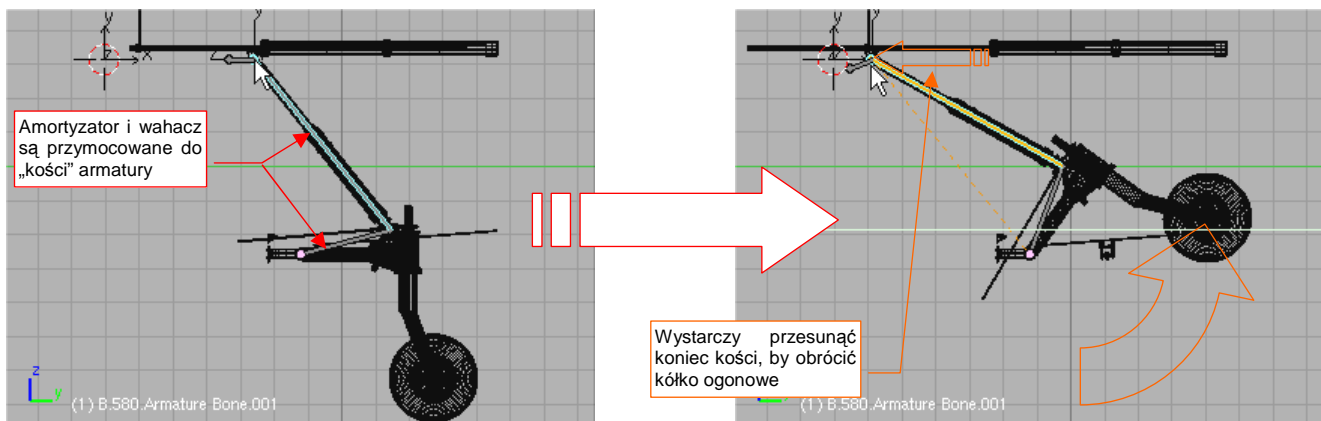
### Podsumowanie

- Podwozie na zwykłych planach samolotu jest zazwyczaj narysowane w zbyt małej skali, a przez to — niewystarczająco dokładnie. Aby jak najlepiej odwzorować je w modelu, zgromadź możliwie dużo zdjęć tego zespołu. Nie zaszkodzi także przygotować jakiegoś schematycznego rysunku (str. 421);
- Przed rozpoczęciem modelowania podwozia dobrze przemyśl, co i jak odwzorujesz. Musisz zdecydować, co należy pominąć, a co — uprościć (str. 424). Mnie pomaga w tym wykonanie odręcznych szkiców modelowanego mechanizmu. Na pewno istnieją także inne metody ☺;
- Najtrudniejszymi do modelowania fragmentami są elementy z wieloma wypustami i otworami. Przykładem tak skomplikowanej części jest np. wahacz kółka ogonowego. Jego formowanie składa się z wielu etapów (str. 424 i dalsze);
- Kształty prawie wszystkich części mechanizmów można „wyprowadzić” z początkowej kombinacji cylindrów i prostopadłościaków. Dobre przybliżenie cylindra uzyskuje się z ośmiokąta, „wygładzonego” modyfikatorem *Subsurf* (z *Level* = 2 dla większych obiektów, lub *Level* = 1 dla mniejszych). Tylko w wyjątkowych wypadkach należy stosować operacje typu *Boolean* (str. 422, 429);
- Inne problemy mogą się kryć w prostych z pozoru przypadkach, jak wyprowadzenie prostopadłej ścianki z okrągłej podstawy (str. 426). Aby coś takiego uzyskać z powierzchni podziałowej (*Subsurf*), musisz przygotować odpowiedni układ wierzchołków. Jeżeli rozpoczniesz siatkę, od razu uwzględniając powstanie takich ostrych krawędzi, nie jest to nawet specjalnie trudne (str. 426);
- Zaraz po stworzeniu nowego obiektu przypisz mu odpowiedni materiał. Do drobnych części używaj prostszych materiałów, których tekstury nie wymagają rozwinięcia siatki w przestrzeni UV. Zazwyczaj taka „stal”, „guma”, czy „ciągną”, będą się później powtarzać w innych zespołach samolotu (str. 422, 423);
- Drobne sprężyny można modelować za pomocą zwykłych walców, pokrytych specjalnym, półprzezroczystym materiałem (str. 429, 432);
- Jako elementy, które się powtarzają (śruby, bolce, płaskowniki, ...) warto wykorzystywać klony pojedynczej siatki (np. wzorcowa śruba — str. 430);
- Czasami trzeba w jakimś konkretnym klonie zmienić materiał (chodzi o sytuacje, gdy ma być inny niż ten, przypisany do wzorcowej siatki). Wystarczy wówczas przełączyć taki element z wykorzystania materiału siatki (*ME*) na materiał obiektu (*OB*) — str. 431;
- Przy wzajemnym dopasowywaniu pokryw luków i mechanizmu podwozia, warto pozostawić oryginalne pokrywy w pozycji „zamkniętej” (tak, jak je wcześniej zamodelowaliśmy). Stwórz za to dodatkowy, roboczy klon pokrywy i jej osi, i ustaw je w pozycji „otwarte” (str. 430);

## 7.2 Mechanizacja kółka ogonowego

Jak już zdążyłeś się przekonać w poprzedniej sekcji, każde podwozie to mniej lub bardziej złożony zespół kół, goleni, amortyzatorów i różnorodnych popychaczy. Większość samolotów z okresu II Wojny Światowej miała w dodatku podwozie chowane. Wyobraź sobie, że robisz dwie sceny ze swoim modelem: jedną na lotnisku, a drugą w locie. W pierwszej podwozie musi być wysunięte, a w drugiej — schowane. Oczywiście, można je chować i wysuwać element po elemencie. Zapewniam Cię jednak, że za drugim razem podobna zabawa staje się już bardzo nużąca. Po co zresztą się tak męczyć, skoro Blender udostępnia nam różnorodne środki animacji? Wystarczy z nich skorzystać, by związać te wszystkie części w jeden mechanizm, uruchamiany przesunięciem pojedynczego uchwytu.

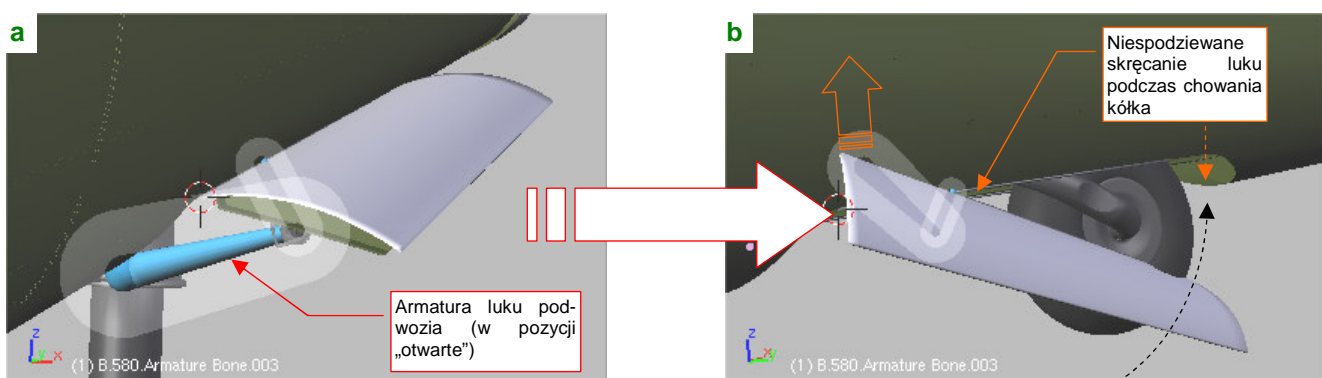
Pierwszym rozwiązaniem, które się w takiej sytuacji nasuwa, jest wykorzystanie takich samych szkieletów, jakich używa się w Blenderze do animacji postaci ludzi czy zwierząt. Te szkielety nazywane są w grafice komputerowej *armaturami* (*armatures*), i składają się (a jakże by inaczej!) — z kości (*bones*). Wstawiający do rysunku nieskomplikowaną armaturę, złożoną z dwóch połączonych kości, można szybko uzyskać podstawowy efekt chowania i wysuwania kółka ogonowego (Rysunek 7.2.1):



Rysunek 7.2.1 Chowanie kółka ogonowego za pomocą armatury

(Wszystko na temat budowy takiej armatury znajdziesz na str. 814 i następnych). Niestety, tak prosto można okiełznać tylko płaski zespół elementów. W przypadku ruchu w przestrzeni kości zaczynają się skręcać wokół osi podłużnej, co prowadzi do zaskakujących efektów.

Zobacz, na przykład, układ dwóch kości, modelujący mechanizm zamykania luku podwozia (Rysunek 7.2.2a):



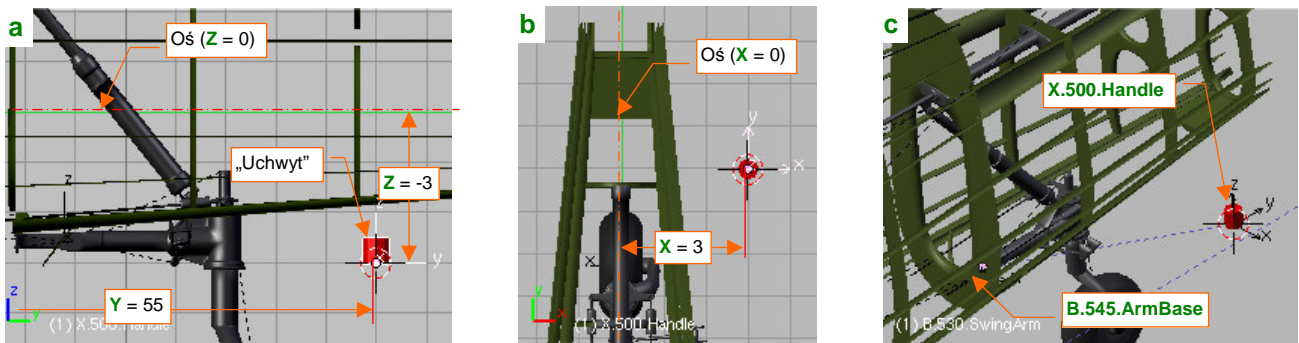
Rysunek 7.2.2 Przykład problemów z odwrotną kinematyką kości (armatura luku podwozia)

Wydaje się równie prosty jak ten poprzedni, prawda? A jednak, gdy koniec ramienia zaczyna wędrować, wraz z wahaczem, do góry, cała pokrywa także się składa, ale jednocześnie — skręca (Rysunek 7.2.2b). Co prawda, można zapanować nad takim efektem ubocznym za pomocą różnych ograniczeń (*Constraints*), ale cała praca zamienia się w dość męczące zajęcie.



Sądzę, że taką „niesforną” armaturę można okiełznać, ale, mówiąc szczerze, nie chciało mi się poświęcać na to czasu. Przy okazji prób radzenia sobie z jej obrotami poznałem system ograniczeń (*constrains*) Blendera. Szybko doszedłem do wniosku, że do osiągnięcia zamierzonego efektu — otwierania i zamykania całego podwozia za pomocą jednego przesunięcia. — kości właściwie nie są potrzebne. Wystarczy tylko zamienić miejscami przyczynę ze skutkiem. To nie wahacz będzie się chował, bo jest ciągnięty w górę przez amortyzator, tylko amortyzator podniesie się do góry, bo jest popychany przez obrót wahacza. Gdy to dobrze rozegramy, zewnętrzny obserwator nawet nie zwróci uwagi, jakie są prawdziwe zależności pomiędzy poszczególnymi częściami. A gdy odwrócisz przyczynę i skutek, możesz cały mechanizm zamodelować za pomocą kilku ograniczeń. Nie trzeba do tego żadnej armatury, ani *Pose Mode*!

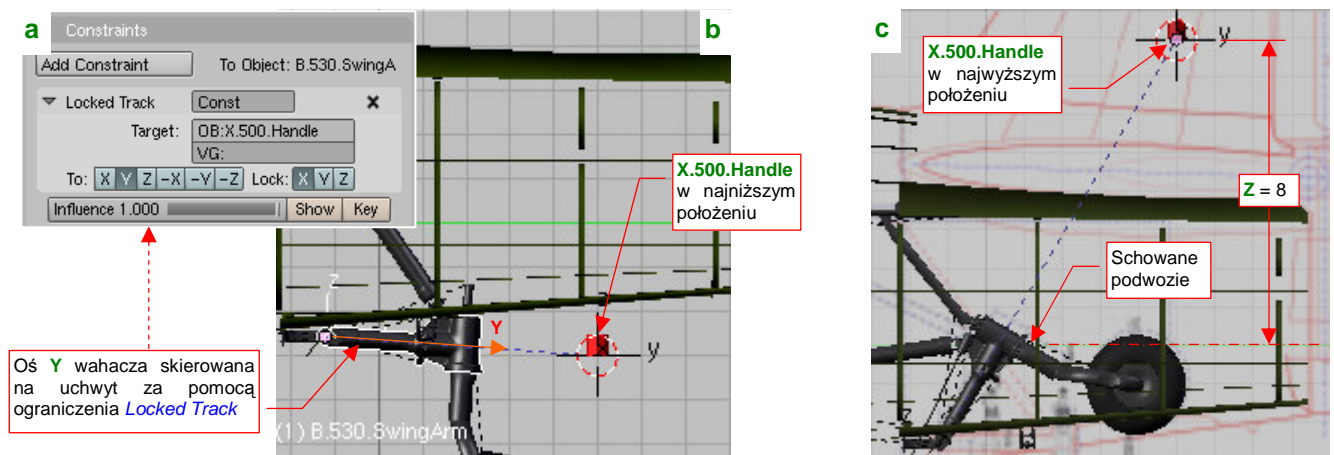
Zacznijmy od wstawienia obiektu — uchwytu, którego przemieszczenie będzie sterować wysuwaniem i chowaniem podwozia (Rysunek 7.2.3):



Rysunek 7.2.3 Uchwyt, który będzie poruszał całym zespołem kółka ogonowego

Osobiście używam na uchwyty zwykłych, ośmiobocznych cylindrów. Przypisałem mu materiał **Artificial** i umieściłem na warstwie **10** (tej samej, na której znajduje się „cel” kamery — por. str. 261). Uchwyt położony jest nieco z boku kadłuba (Rysunek 7.2.3a,b) by w razie potrzeby łatwo go było dostrzec. Postaraj się dobrać jego położenie tak, by jego współrzędne były liczbami całkowitymi. Nie jest to niezbędne, ale bardzo ułatwia dalszą pracę. Nadajmy mu nazwę z innym przedrostkiem, by odróżniał się od „zwykłych części”. Niech to będzie **X.500.Handle**. Włącz uchwyt do hierarchii modelu, przypisując go (relacją *Parent*) do okucia wahacza (**B.545.ArmBase** — por. Rysunek 7.2.3c, a także str. 430).

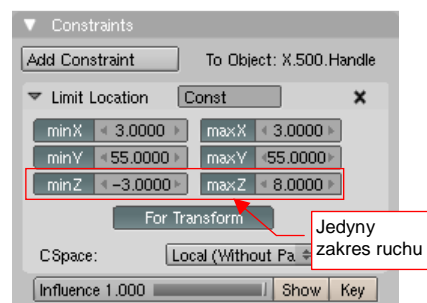
Aby wahacz (**B.530.SwingArm**) podążał za ruchem uchwytu, przypisz mu ograniczenie *Locked Track* (Rysunek 7.2.4, więcej na temat tego ograniczenia — str. 823):



Rysunek 7.2.4 Przypisanie do wahacza ograniczenia *Locked Track*.

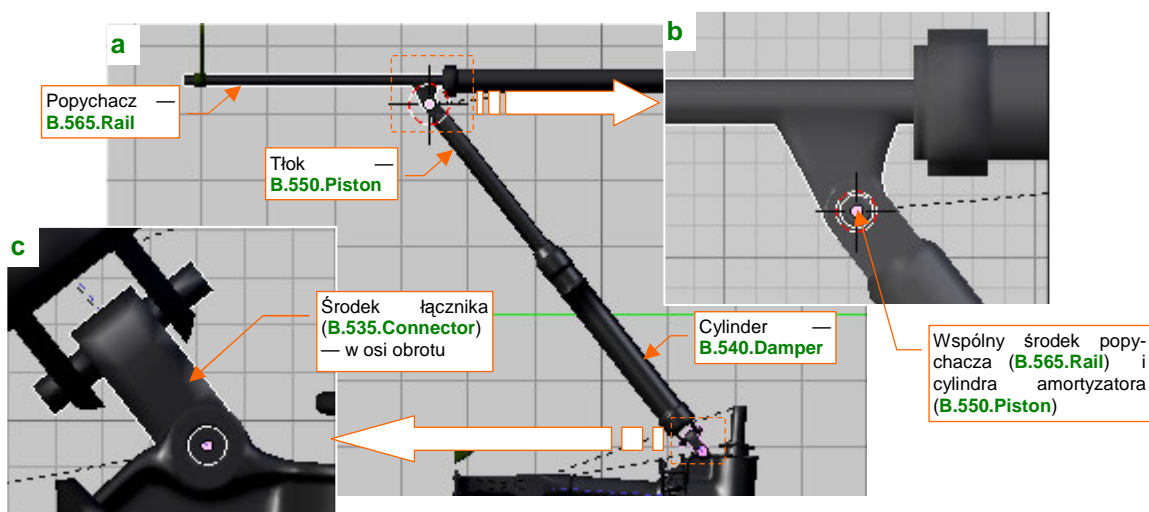
W parametrach ograniczenia wskaż jako cel obiekt **X.500.Handle** (Rysunek 7.2.4a). Ustal **X** jako oś obrotu, a **Y** niech celuje w środek uchwytu (Rysunek 7.2.4b). Gdy przesuniesz uchwyt do góry (do pozycji **Z = 8.0**), koło podwozia znajdzie się w pozycji „schowane” (Rysunek 7.2.4b).

Dolne i górne położenia uchwytu musisz dopasować tak, by odpowiadały właściwemu położeniu kółka ogonowego. (Położenie dolne powinno odpowiadać podwoziu całkowicie wysuniętemu, a górne — schowanemu). W razie czego możesz lepiej dobrać to położenie, zmieniając odległość uchwytu od środka obrotu wahacza (zmieniając współrzędną  $Y$ ). Gdy zakres ruchu uchwytu jest już ustalony, „utwral” go za pomocą odpowiedniego ograniczenia *Limit Location* (Rysunek 7.2.5). (Więcej na temat tego rodzaju ograniczenia i jego parametrów znajdziesz na str. 827.



Rysunek 7.2.5 Ograniczenie ruchu uchwytu.

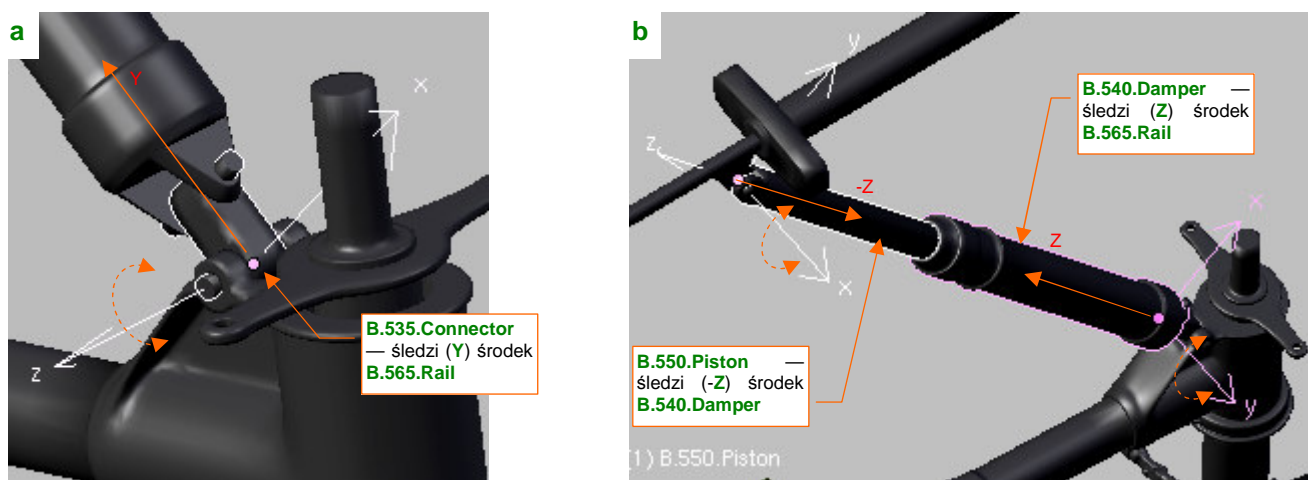
Zajmijmy się teraz górną częścią podwozia: popychaczem i amortyzatorem. Bardzo ważne jest odpowiednia pozycja środków i hierarchii tych elementów (Rysunek 7.2.6):



Rysunek 7.2.6 Położenie środków poszczególnych części zespołu amortyzatora

Środek popychacza (**B.565.Rail**) umieść (jak — zob. str. 808) w osi obrotu tłoka amortyzatora (Rysunek 7.2.6b). Jeżeli chodzi o hierarchię, to przypisz (relacją *Parent*) tłok (**B.550.Piston**) do popychacza **B.565.Rail**. Cylinder amortyzatora (**B.540.Damper**) przypisz (relacją *Parent*) do niepozornego łącznika (**B.535.Connector** — Rysunek 7.2.6c), a sam łącznik — do wahacza (**B.530.SwingArm**).

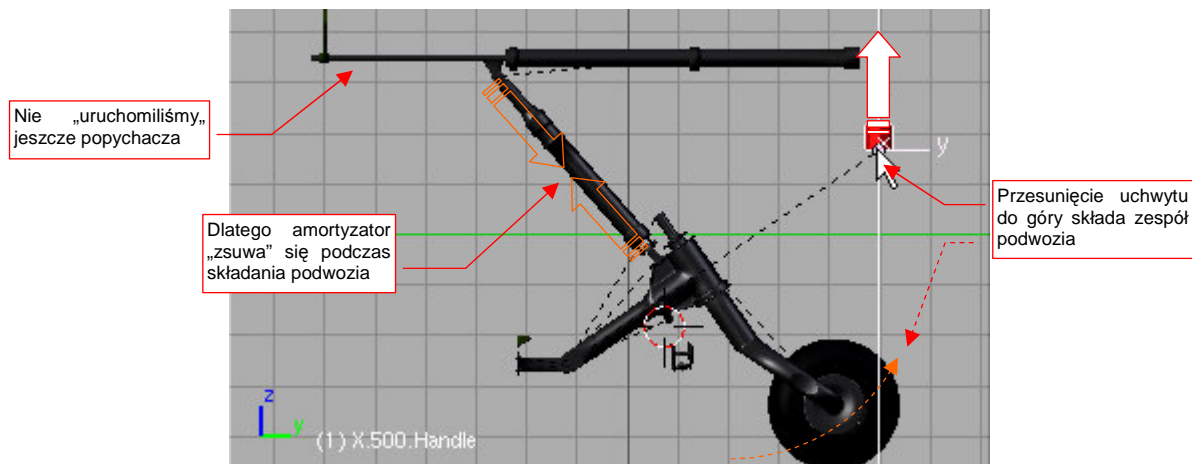
Teraz przypisz łącznikowi **B.535.Connector** ograniczenie *Locked Track* tak, by jego oś  $Y$  „celowała” w środek popychacza (Rysunek 7.2.7a):



Rysunek 7.2.7 Ograniczenia *Locked Track*, przypisane do elementów amortyzatora

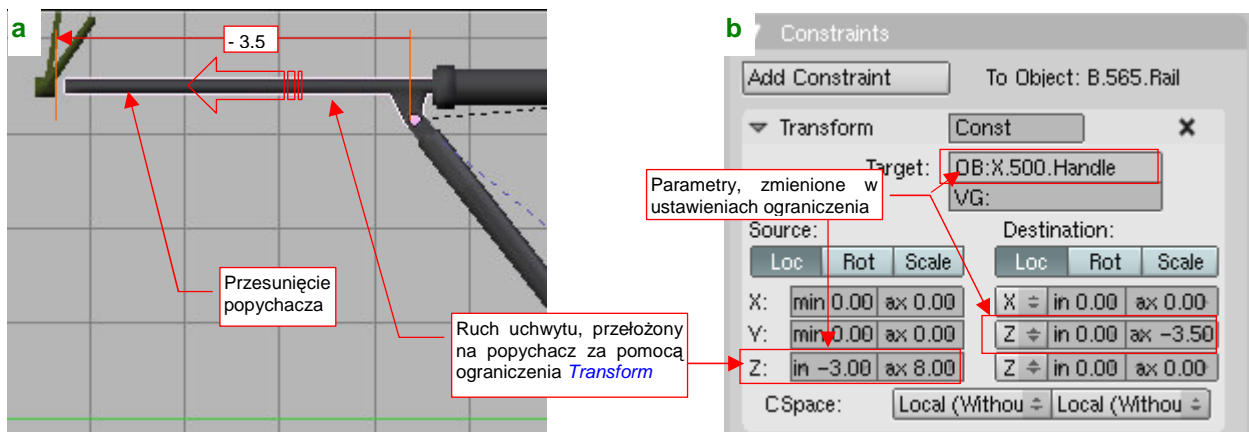
Jednocześnie ogranicz cylinder **B.550.Piston** tak, by jego oś  $Z$  śledziła środek cylindra **B.540.Damper** (Rysunek 7.2.7b). Na wszelki wypadek możesz także dodatkowo ograniczyć cylinder, by „celował” w tłok.

Na razie nasze ograniczenia powodują, że w trakcie obrotu wahacza amortyzator jest niemiłosiernie ściskany (Rysunek 7.2.8):



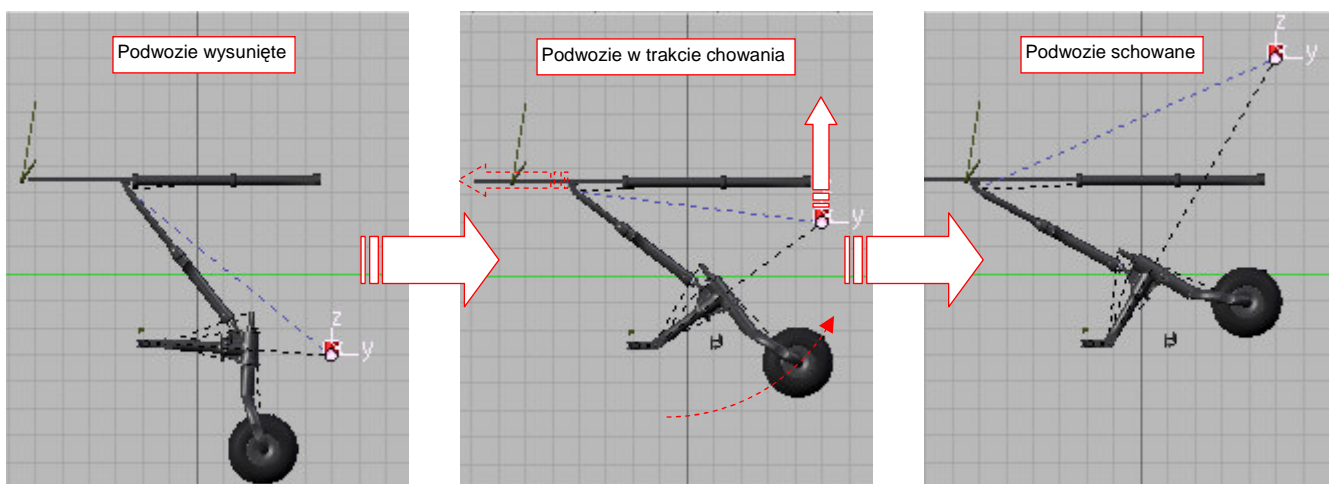
Rysunek 7.2.8 Rezultat działania ograniczeń: ugięcie amortyzatora

To dlatego, że nie uruchomiliśmy jeszcze popychacza. Gdy będziesz przesuwał uchwyt **X.500.Handle** do góry, popychacz powinien się przesuwać w lewo (Rysunek 7.2.9a). Najprościej ten efekt uzyskać za pomocą „uniwersalnej przekładni” — ograniczenia **Transform** (Rysunek 7.2.9b):



Rysunek 7.2.9 Przełożenie ruchu uchwytu na przesunięcie popychacza (ograniczenie **Transform**)

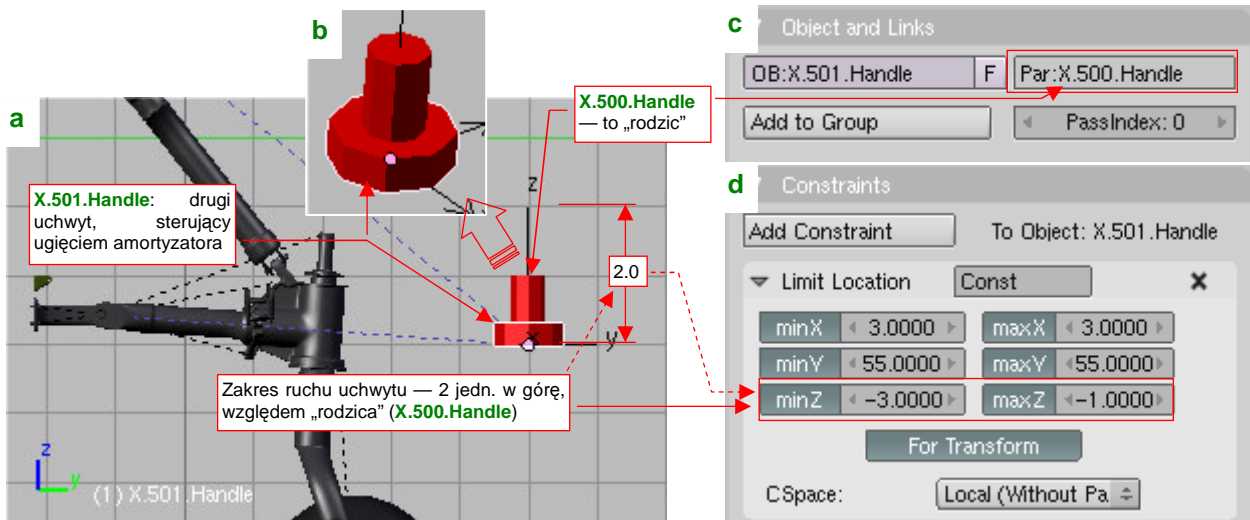
(Szczegółowy opis parametrów i działania ograniczenia **Transform** znajdziesz na str. 829). Dystans przesunięcia popychacza — 3.5 jedn. — dopasowałem po kilku „przymiarkach”. W ten sposób uzyskaliśmy już całkiem poprawny ruch mechanizmu podwozia (Rysunek 7.2.10):



Rysunek 7.2.10 Chowanie podwozia za pomocą uchwytu **X.500.Handle**.

Do tej pory ustawialiśmy koło w pozycji wysuniętej tak, jak wyglądało tuż przed lądowaniem — gdy amortyzator był zupełnie bez obciążenia. Jednak wtedy, gdy samolot stoi na ziemi, amortyzator „zsuwa” się trochę. Powoduje to odchylenie wahacza.

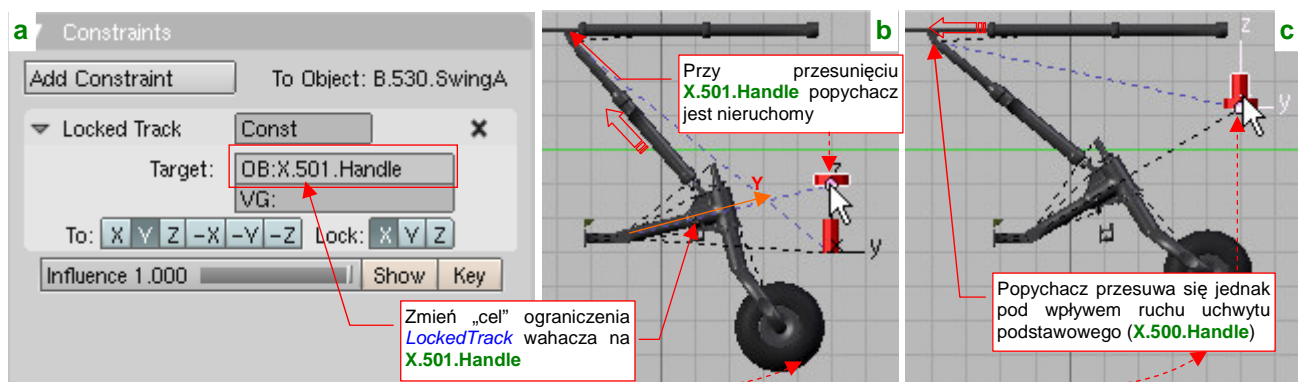
Aby odwzorować takie ugięcie podwozia, „nałoż” na uchwyt **X.500.Handle** drugi, mniejszy (ale szerszy! — Rysunek 7.2.11b). Nadajmy mu nazwę **X.501.Handle** (Rysunek 7.2.11a):



Rysunek 7.2.11 Dodatkowy uchwyt, sterujący ugięciem amortyzatora (X.501.Handle).

Przypisz (relację *Parent*) uchwyt **X.501.Handle** do podstawowego uchwytu (**X.500.Handle**) (Rysunek 7.2.11c). Dzięki temu uchwyt pomocniczy będzie „przyczepiony” do **X.500.Handle**, gdy zaczniesz nim poruszać, by schować podwozie. Aby ugięcie amortyzatora nie było „przeięciem”, dodaj do uchwytu **X.501.Handle** ograniczenie *Limit Location* (Rysunek 7.2.11d). Pozwól przesuwac mu się wyłącznie w pionie (wzdłuż lokalnej osi **Z**), na dystans 2 jednostki (od -3.0 do -1.0). (Po kilku przymiarkach zdecydowałem, że taki zakres będzie odpowiedni — powoduje „zsuniecie” amortyzatora o około 1 jednostkę Blendera. Odpowiada to 10 cm na rzeczywistym samolocie).

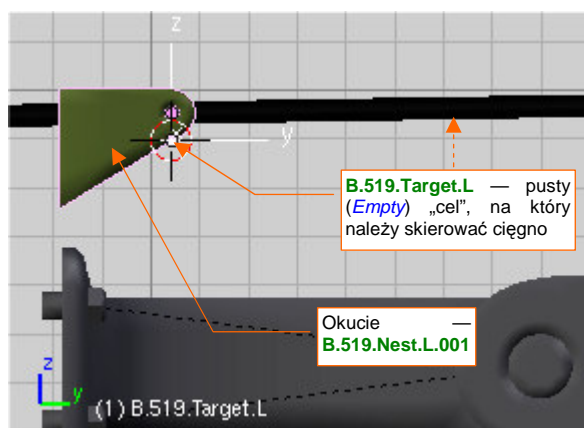
Teraz zmodyfikuj „cel” ograniczenia *Locked Track* wahacza (**B.530.SwingArm**). Wpisz tam, w miejsce nazwy uchwytu podstawowego, uchwyt pomocniczy (**X.501.Handle**) (Rysunek 7.2.12a):



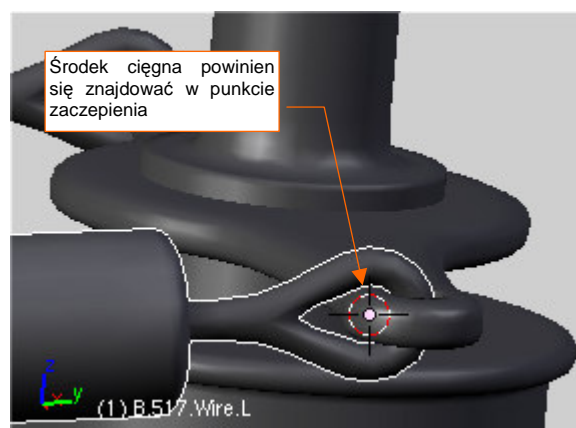
Rysunek 7.2.12 Sterowanie ugięciem amortyzatora.

Początkowo nie zmieni to niczego, gdyż środki obydwu uchwytów się pokrywają (por. Rysunek 7.2.11a). Gdy jednak zaczniesz przesuwać **X.501.Handle** do góry, zauważysz że ten ruch wywołuje nasunięcie na tłok cylindra amortyzatora — tak, jak gdyby był obciążony (Rysunek 7.2.12b). Jednocześnie — popychacz zacznie się wysuwać tak jak to sprawdzaliśmy poprzednio, gdy przesuńiesz do góry uchwyt podstawowy (**X.500.Handle**) (Rysunek 7.2.12c). Zwróć uwagę, że **X.501.Handle** porusza się wraz z nim. To dzięki ich przypisaniu (w hierarchii przedmiotów — relacją *Parent*). Musisz tylko pamiętać, by przed zamknięciem podwozia przesunąć uchwyt pomocniczy zupełnie do dołu (tj. ustawić koło w pozycji „swobodnej”).

Kolejnym elementem „do ograniczania” są cięgna. Zaczynij od wstawienia w każde z okuc — lewe i prawe — pomocniczych obiektów typu *Empty* (por. str. 791). Nadaj im nazwę **B.519.Target.L** i **.R** (Rysunek 7.2.13):



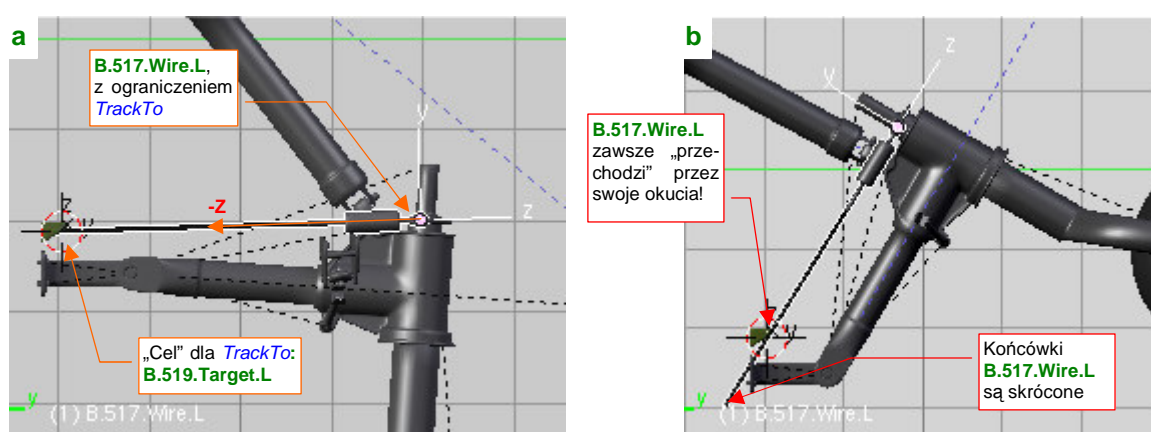
Rysunek 7.2.13 Pusty obiekt (*Empty*) — cel dla cięgna



Rysunek 7.2.14 Położenie środka cięgna

Jeżeli chodzi o strukturę modelu — przypisz (relacją *Parent*) każdy z tych „celów” do odpowiedniego okucia. (**B.519.Target.L** do **B.519.Nest.L.001**, a **B.519.Target.R** do **B.519.Nest.R.001**)

Zajmijmy się teraz cięgnami. Zwróć uwagę na ich środki — powinny się znajdować w miejscu ich zaczepienia (Rysunek 7.2.14). Gdy już tam są, możesz do każdego cięgna dodać ograniczenie *Track To* (por. str. 821). Jako „cel” (*Target*) tych ograniczeń wpisz odpowiedni z obiektów **B.519.Target** (**.L** lub **.R**) (Rysunek 7.2.15a):



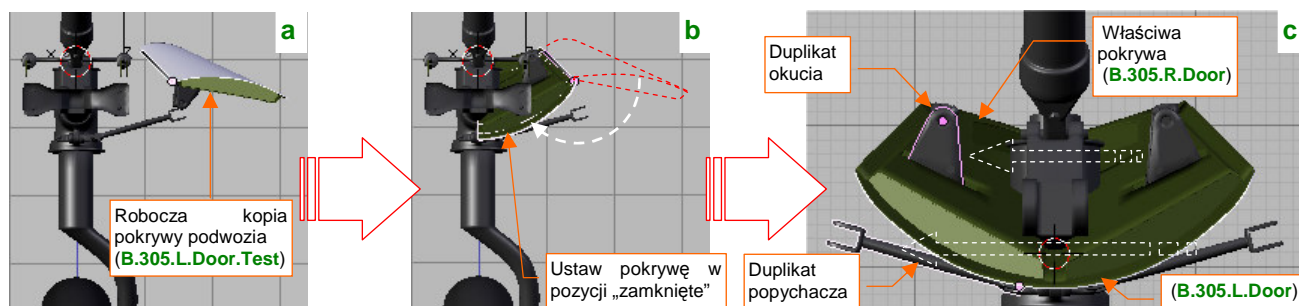
Rysunek 7.2.15 „Układanie się” cięgien podczas ruchu podwozia

- W przypadku cięgien należy stosować ograniczenie *Track To*, a nie *Locked Track*, gdyż mogą się obracać względem dwóch osi. (*Locked Track* imituje możliwość obrotu wokół jednej osi)

Za pomocą ograniczeń *Track To* sprawiłeś, że w każdej pozycji podwozia cięgna będą „przechodzić” przez własne okucia. W przyszłości, aby podnieść realizm modelu, możesz przełożyć ruch steru kierunku na skręcenie goleni kółka ogonowego. (Oryginalnie cięgna **B.517.Wire** służyły właśnie do tego). W naszym modelu nie będzie to przyczyną, a skutkiem ruchu: w wyniku obrotu goleni obróci się przypisany do niej orczyk. To z kolei spowoduje niesymetryczne przesunięcie środków cięgien (jedno się cofnie, drugie przesunie do przodu). Obydwa zbliżą się nieznacznie do osi kadłuba (poruszają się po okręgu). Dzięki ograniczeniu *Track To*, nawet przy takim przemieszczeniu ich końce będą przechodzić przez okucia, przymocowane do wręgi kadłuba.

Na koniec warto skrócić końcówki tych drutów, aby przy całkowicie schowanym podwoziu nie „sterczały” z poszycia (Rysunek 7.2.15b). Na szczęście można dobrać ich długość tak, by jednocześnie przy zupełnie wysuniętym kole „dotykały” powierzchni wręgi. Kłopot może się zacząć, gdy obrócisz orczyk — wtedy końcówka jednego z nich stanie się widoczna! Może trzeba tu będzie potem „dosztukować” jakiś pomocniczy „drut”? Udawałby dalszą część cięgna, ale poruszałby się tylko poziomo. Byłby wysuwany tylko przy obrocie orczyka.

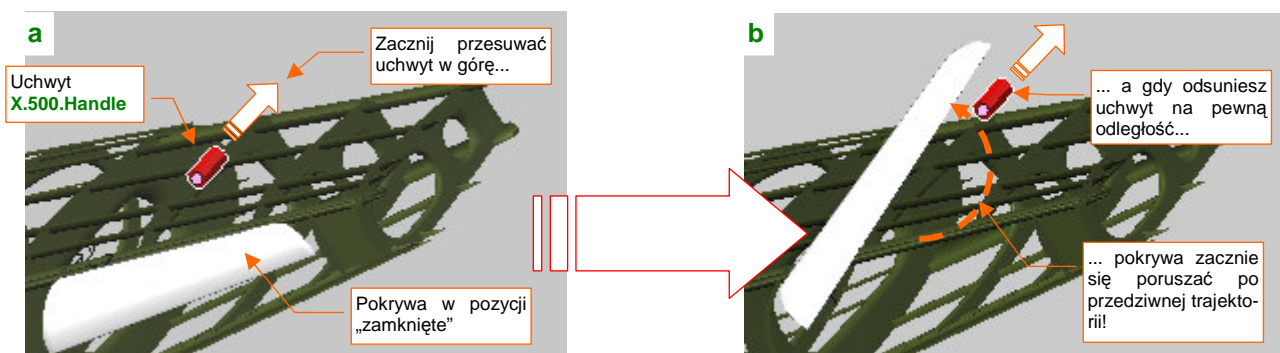
Ruch wahacza powodował w P-40 otwieranie lub zamykanie pokryw podwozia. Włączmy je więc w wirtualny mechanizm, który tu budujemy. W poprzedniej sekcji zamodelowaliśmy okucie i popychacz tylko dla lewej pokrywy. Umieściliśmy je na jej „roboczej” kopii — **B.305.L.Door.Test** (por. str. 430). Włącz teraz widoczność warstwy, na której znajduje się ten obiekt (Rysunek 7.2.16a):



Rysunek 7.2.16 Powielenie okuc i popychacza pokryw podwozia

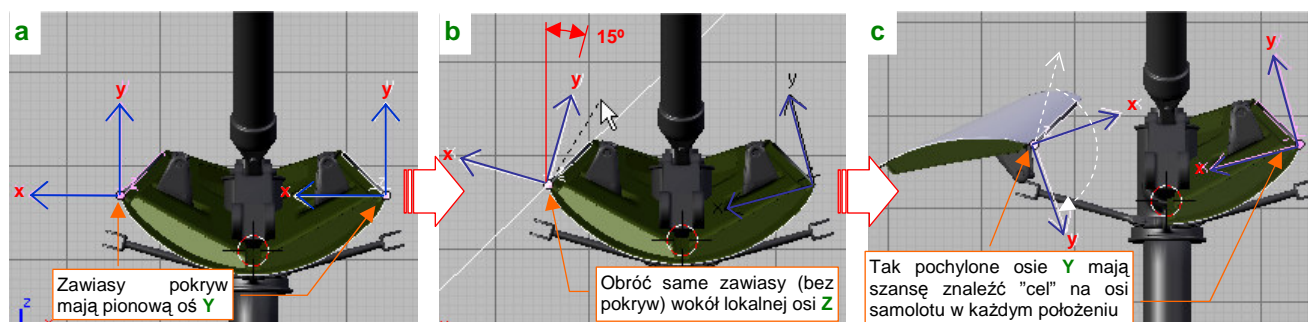
Obróć pokrywę wokół lokalnej osi **Z** jej zawiasów, tak, by przyjęła pozycję „zamkniętą”. W ten sposób okucie znalazło się w położeniu dopasowanym do „właściwych” pokryw — **B.305.L.Door** i **B.305.R.Door** — których nigdy dotąd nie ruszyliśmy z pozycji „zamkniętej”. Odłącz je od roboczej kopii (**Alt-P** — *Clear Parent*) i przypisz do pokrywy **B.305.L.Door**. Jego lustrzane odbicie przypisz do **B.305.R.Door**. (Szczegółowy opis tej operacji — jak stworzyć lustrzane odbicie okucia, jak przypisywać je do pokryw — znajdziesz na str. 838). Potem możesz już usunąć cały zespół roboczej pokrywy **B.305.L.Door.Test** — nie będziemy go więcej potrzebować.

Ruchu uchwyty nie da się dobrze przełożyć na obrót pokrywy za pomocą *Transform* (Rysunek 7.2.17):



Rysunek 7.2.17 Nieoczekiwany efekt przełożenia ruchu uchwyty na obrót pokrywy za pomocą *Transform*.

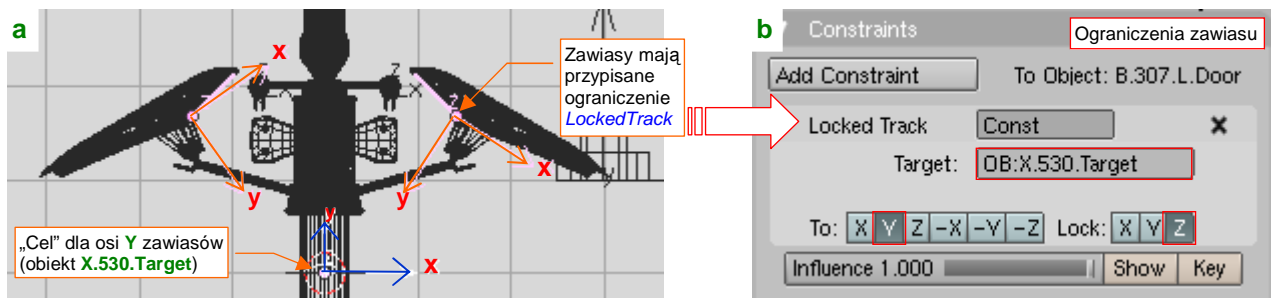
Szczegółowe wyjaśnienie, dlaczego pojawił się efekt z Rysunek 7.2.17b), znajdziesz na str. 831. A nam pozostaje wywołać obrót pokryw za pomocą ograniczenia *Locked Track*. Nim jednak przypiszemy je zawiasom, trzeba trochę pochylić ich osie **Y** — bo obecnie są pionowe, a to uniemożliwia ich domknięcie (Rysunek 7.2.18a):



Rysunek 7.2.18 Zmiana orientacji zawiasów pokryw podwozia

Odłącz na chwilę pokrywę od zawiasów (**Alt-P**). Obróć zawiasy o  $15^\circ$  wokół lokalnych osi **Z** (Rysunek 7.2.18b). Potem z powrotem przypisz pokrywę do zawiasów. Teraz w całym zakresie położenia osie **Y** przecina się z płaszczyzną symetrii samolotu (Rysunek 7.2.18c) — więc można tam umieścić „cel” dla ograniczenia *Locked Track*.

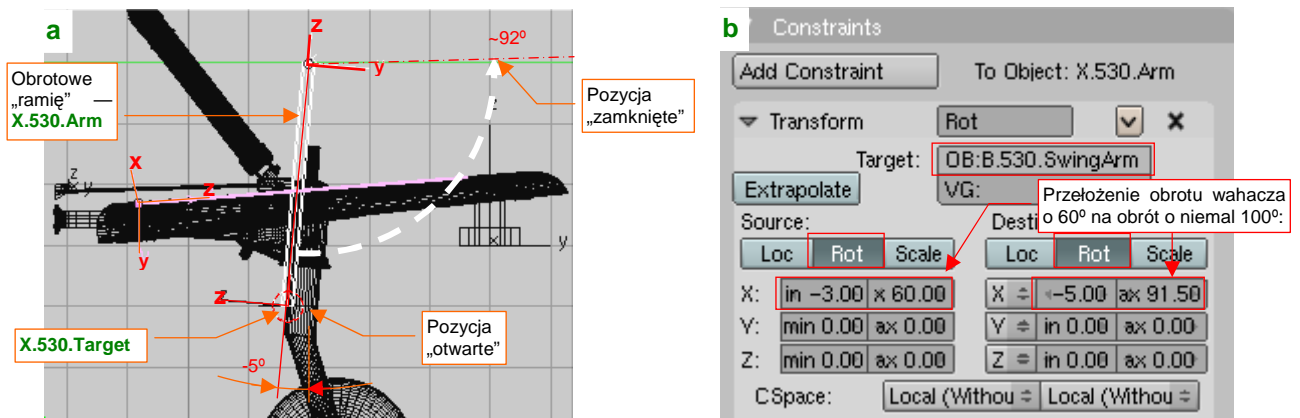
Wstaw w płaszczyźnie symetrii modelu „cel” dla osi **Y** zawiasów: pusty (*Empty*) obiekt o nazwie **X.530.Target** (Rysunek 7.2.19a):



Rysunek 7.2.19 Skierowanie zawiasów na pomocniczy obiekt — cel (ograniczenie *Locked Track*)

Następnie każdemu z zawiasów przypisz odpowiednie ograniczenia *Locked Track* — z zablokowaną osią **Z** (Rysunek 7.2.19b). Sam cel tego ograniczenia — **X.530.Target** — umieść na takiej wysokości, by pokrywy odchyliły się jak w pozycji „otwarte” ( $Z = -4.0$ ).

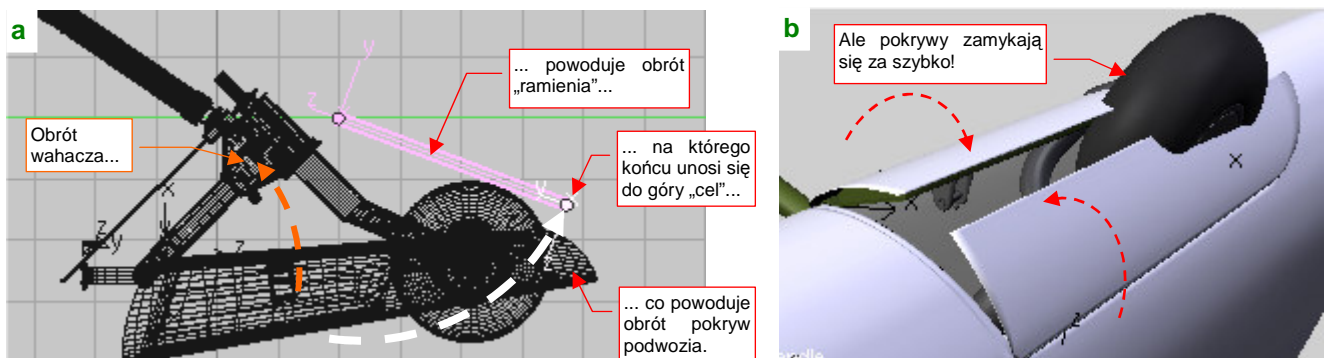
A jak wywołać ruch tego układu? „Zamocuj” (relacją *Parent*) obiekt — „cel” (**X.530.Target**) na obrotowym ramieniu (Rysunek 7.2.20a):



Rysunek 7.2.20 Ramie, po którym porusza się obiekt — cel (przełożenie obrotu wahacza ograniczeniem *Transform*)

Stwórz je jako „pręt”, umieszczony — jak wszystkie obiekty pomocnicze — na warstwie 9. Nadaj mu nazwę **B.530.Arm** i przypisz (relacją *Parent*) do kadłuba. Środek (oś obrotu) jest na końcu tego pręta, w punkcie  $Y = 51.0$ ,  $Z = 0.0$ . Długość — 4.0. Na przeciwległym krańcu znajduje się „cel” — **X.530.Target**. Równie dobrze mógłby to być obiekt typu *Empty*, ale uważam, że lepiej jest użyć w tej roli coś bardziej „namacalnego”.

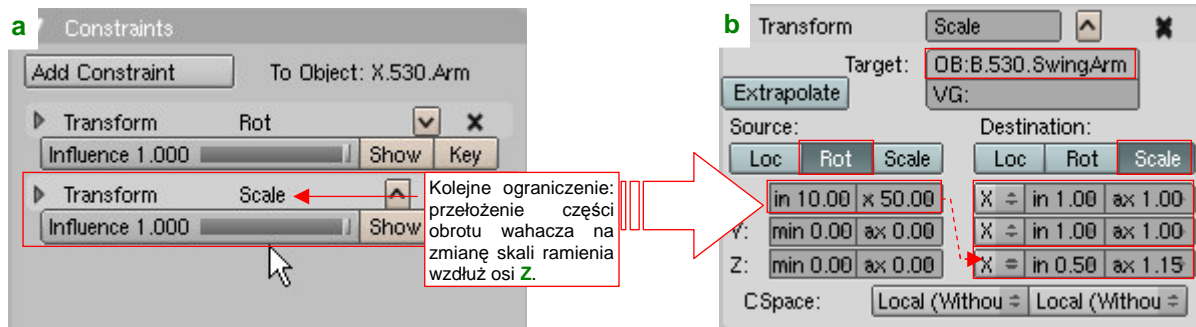
Obrót **X.530.Arm** jest powiązany z obrotem wahacza **B.530.SwingArm** ograniczeniem *Transform* (Rysunek 7.2.20b). Zakres ruchu — od  $-5^\circ$  do około  $92^\circ$ . Ramie obraca się szybciej od wahacza (Rysunek 7.2.21a):



Rysunek 7.2.21 Problem: pokrywy zamykają się zbyt szybko!

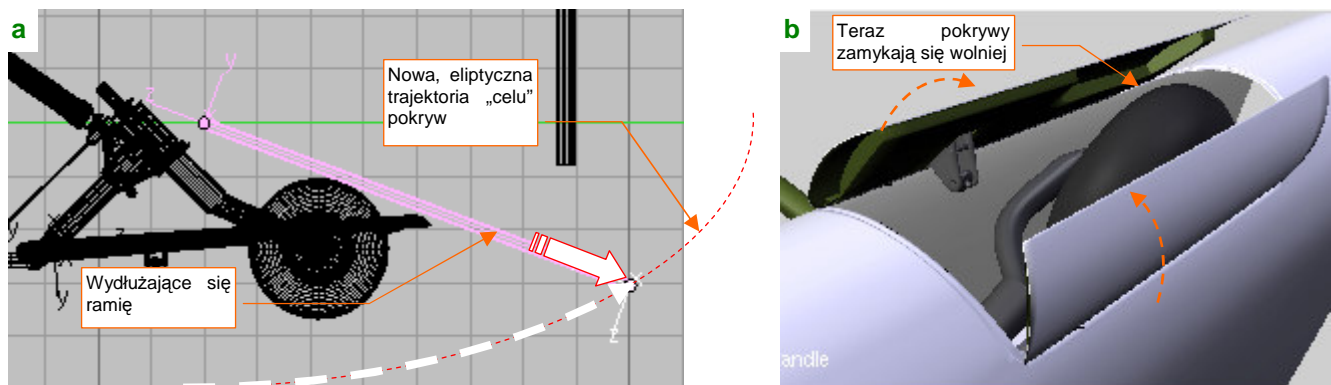
Choć na koniec koło poprawnie zniknie pod pokrywami, to jednak w trakcie ruchu pokrywy zamykają się zbyt szybko i w końcowej fazie „przycinają” oponę i piastę (Rysunek 7.2.21b).

To „przycinanie” koła pokazuje, że prawdziwa trajektoria „celu” dla osi **Y** zawiasów nie może być okręgiem (takim, jaki pokazuje Rysunek 7.2.21a). Dokładne odwzorowanie byłoby zbyt pracochłonne, więc przybliżymy je za pomocą drugiego ograniczenia, zmieniającego długość (skalę wzdłuż osi **Z**) ramienia **X.530.Arm**:



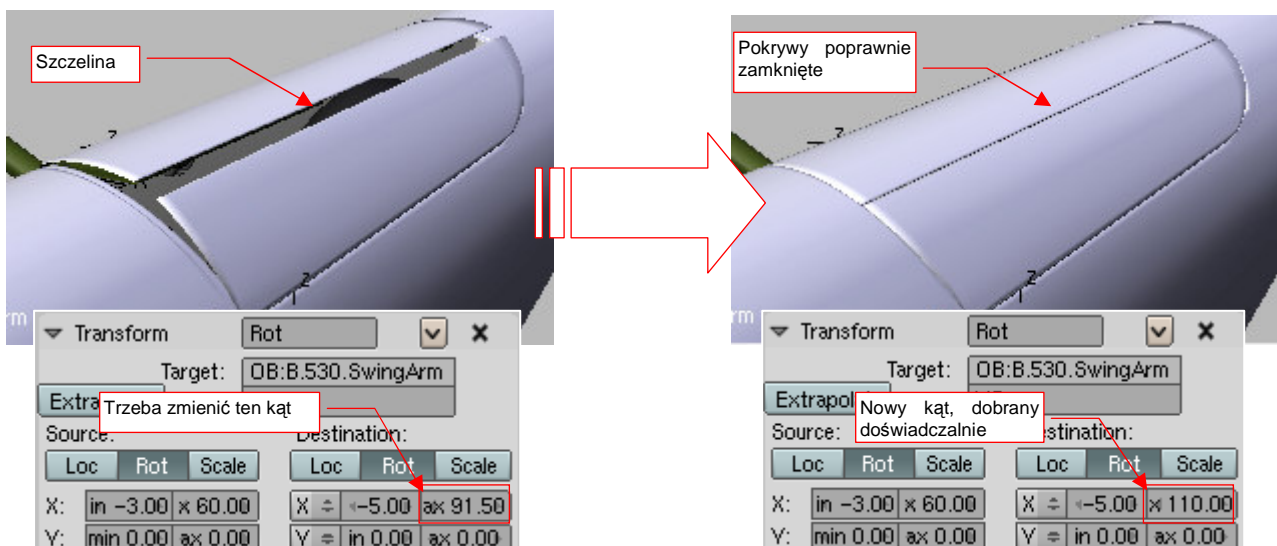
Rysunek 7.2.22 Dodanie do ramienia dodatkowej zmiany skali

Do „stosu” ograniczeń ramienia **X.530.Arm** dodaj drugie, o nazwie **Scale** (Rysunek 7.2.22a). Źródło (**Target**, **Source**) transformacji jest to samo — obrót wahacza **B.530.SwingArm**. Tym razem podczas „środkowego” etapu ruchu (kąąt pochylenia wahacza od  $10^\circ$  do  $50^\circ$ ) ramię stopniowo się wydłuża — z 0.50 (dotychczasowa skala) do 1.15. W efekcie trajektoria „celu” pokryw zmienia się z okręgu na elipsę (Rysunek 7.2.23a):



Rysunek 7.2.23 Dodanie do ramienia dodatkowej zmiany skali

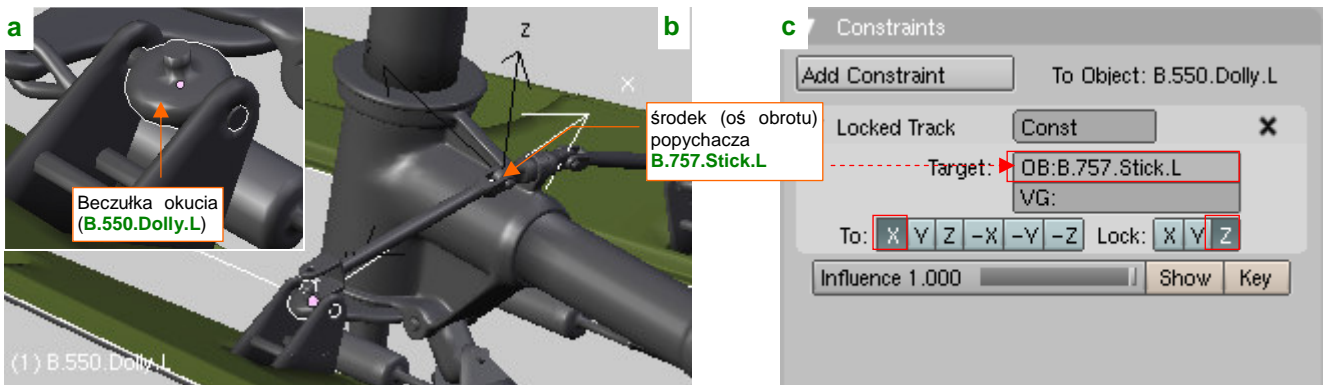
W wyniku dynamicznego wydłużania ramienia, dla tego samego kąta wahacza „cel” jest znacznie niżej (por. str. 441, Rysunek 7.2.21). Dzięki temu pokrywy są szerzej otwarte i nie „przycinają” kółka ogonowego (Rysunek 7.2.23b). Pozostało jeszcze tylko dobrać końcowy kąt pochylenia **X.530.Arm** (w ograniczeniu **Rot**). Zrób to tak, by nie pojawiała się pomiędzy pokrywami żadna szczelina (Rysunek 7.2.24):



Rysunek 7.2.24 „Domknięcie” pokryw — dopasowanie końcowego kąta ramienia **X.530.Arm**



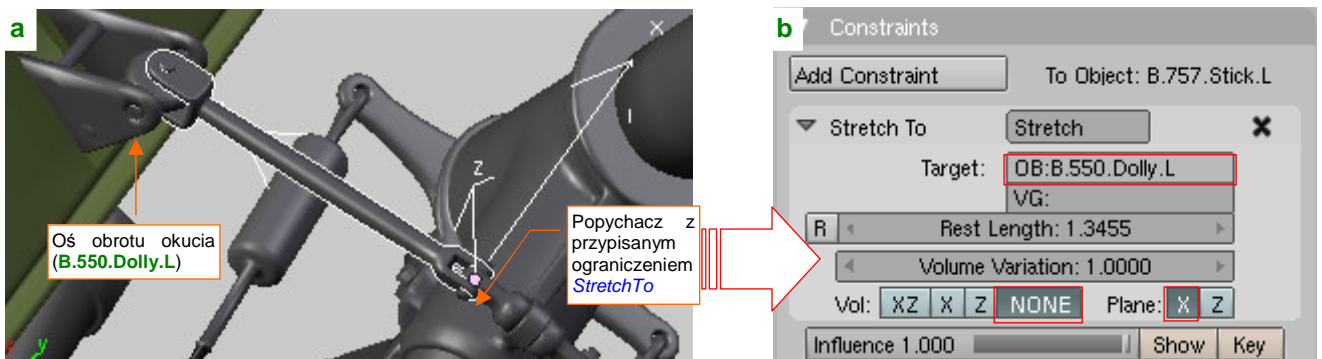
Ostatnim elementem, który pozostał nam do mechanizacji, są popychacze pokryw podwozia. Zaczniemy od beczulek okucia, do których są przymocowane (Rysunek 7.2.25a):



Rysunek 7.2.25 Skierowanie beczułki okucia na oś popychacza (**Locked Track**)

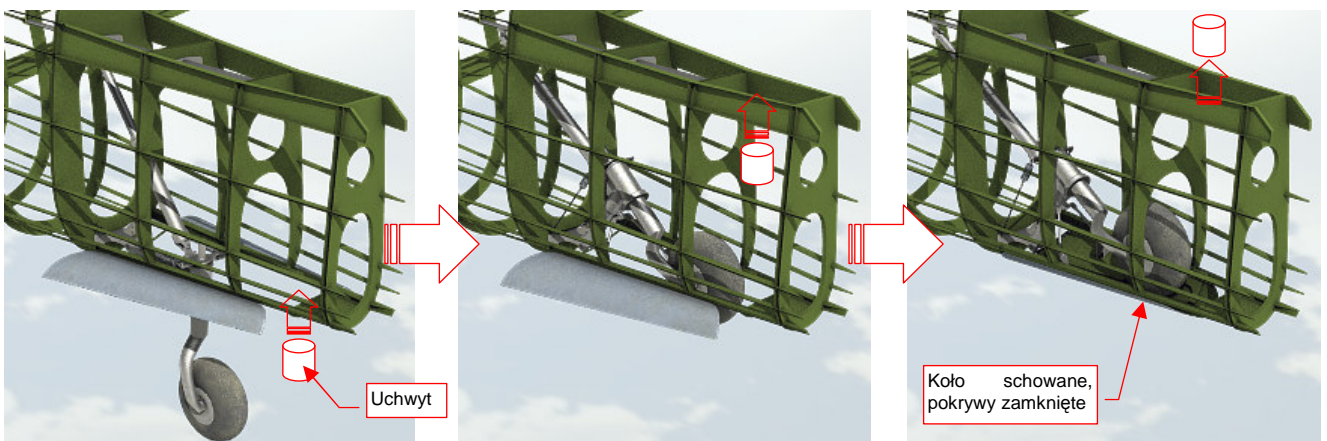
Przypisz beczułkom ograniczenia **Locked Track**. Skieruj ich osie **X** na środki (*origin*) popychaczy (Rysunek 7.2.25c). (Te punkty znajdują się na wewnętrznych krańcach popychaczy — por. Rysunek 7.2.25b).

Jednocześnie skieruj osie **Y** popychaczy na beczułki okuć (**B.550.Dolly.L** lub **.R**). Tutaj jednak nie posłużymy się ograniczeniem **Locked Track**. Gdybyśmy to zrobili, to pomiędzy końcami popychaczy i beczułkami pojawiałyby się wyraźne odstępy (por. str. 833). (Jest to rezultat przybliżenia rzeczywistego ruchu pokryw podwozia za pomocą elipsy — por. str. 442). Zamiast tego wykorzystamy inny rodzaj ograniczenia: **Stretch To** (Rysunek 7.2.26):



Rysunek 7.2.26 Dopasowanie popychacza do okucia (**Stretch To**)

Szczegółowe informacje o działaniu i parametrach tego ograniczenia znajdziesz na str. 833. Dzięki **Stretch To** końce popychaczy będą się trzymać beczulek „jak przyklejone”. Rysunek 7.2.27 przedstawia ostateczne działanie mechanizmu, który stworzyliśmy w tej sekcji:



Rysunek 7.2.27 Składanie kółka ogonowego — jednym przesunięciem uchwytu

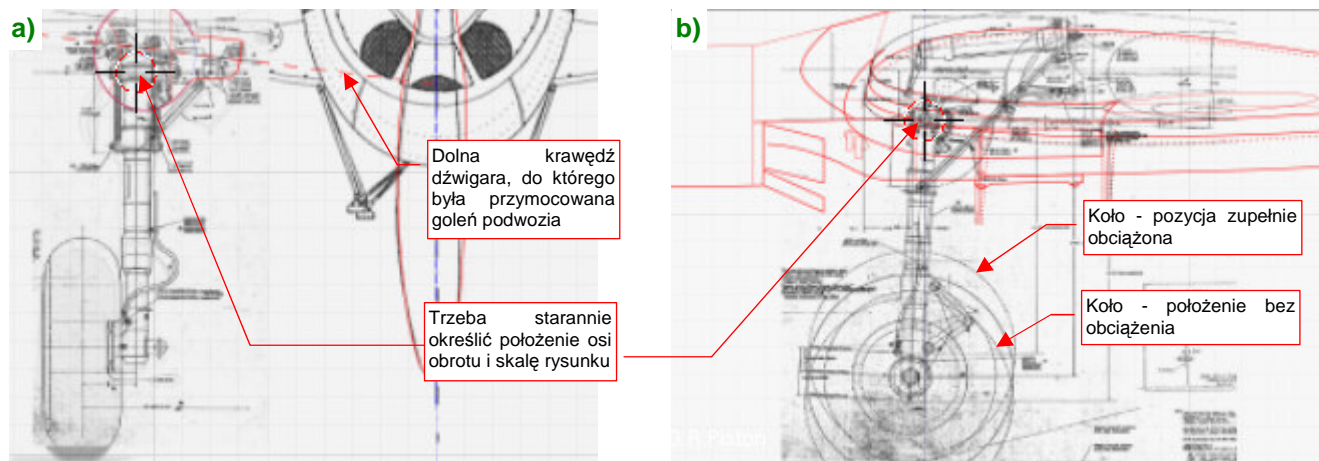
W niektórych stadiach chowania podwozia popychacze, w wyniku działania *Stretch To*, rozciągają się jak gdyby były z gumy. To kompromis, z którym musiałem się pogodzić. Najważniejsze jest, by podwozie wyglądało poprawnie w pozycji „otwarte”. W pozycji „zamknięte” nie powinno razić, choć może odbiegać od oryginału w nieznaczących szczegółach (nikt pod poszycie nie będzie zaglądał). Układ mechanizmu podczas chowania podwozia nie odpowiada dokładnie oryginałowi (pokrywy zamykają się zbyt wolno), ale przynajmniej — nie razi.

### Podsumownie

- „Mechanizacja” polega na połączeniu poszczególnych części modelu w jedną całość (mechanizm). Chodzi o to, by można było łatwo uruchomić model jakiegoś zespołu — np. podwozia — za pomocą jednego przesunięcia jakiegoś pomocniczego uchwytu (str. 434);
- Ruch i wzajemne zależności części mechanizmu można najwierniej odwzorować za pomocą tzw. armatury (*Armature*), zawierającej zespół tzw. „kości” (*bones*). Niestety, w przypadku takich elementów jak podwozie, próba zastosowania kości może się wiązać z poważnymi trudnościami (str. 434);
- W tej sekcji przedstawiam metodę alternatywną: zamiast używać armatury i jej kości, można zbudować cały „wirtualny” mechanizm przypisując elementom zespołu różnego typu ograniczenia ruchu (*constraints*);
- Pracę zaczynamy od stworzenia na warstwie 10 łatwo dostępnego „uchwytu”. Ruch tej części będzie poruszał całym mechanizmem (str. 435);
- Podstawowym ograniczeniem, imitującym zawiasy i inne połączenia osiowe, jest *Locked Track* (str. 435). Przed zastosowaniem tego ograniczenia warto jest zadbać o odpowiednią orientację osi — by w ogóle miały szansę odpowiednio „wycelować” (str. 440);
- Ograniczenie *Limit Location* pozwala ustalić zakres ruchu uchwytu — a w rezultacie całego mechanizmu — do zakresu położenia odpowiadającego rzeczywistości (str. 436);
- Ograniczenie *Transform* można traktować jako „uniwersalną przekładnię” ruchu jednego przedmiotu na inny. Można jej użyć, by uzyskać obrót, czy nawet zmianę skali, z przesunięcia jakiegoś wzorcowego obiektu (str. 437). Tego ograniczenia nie należy stosować do przełożenia skomplikowanego ruchu — na przykład jednoczesnego obrotu wokół wszystkich trzech osi (str. 440);
- „Zsuwający” się amortyzator należy zamodelować z dwóch części: cylindra i tłoka. Każda z nich powinna mieć środek obrotu na przeciwnym krańcu, i innego „rodzica” (*parent*). Tłok powinien być „celem” (*Target*) ograniczenia *Locked Track* cylindra, a cylinder — tłoka (str. 436);
- Ugięcie amortyzatora pod obciążeniem jest osobnym ruchem mechanizmu. Dlatego można je uzależnić od dodatkowego uchwytu (str. 438);
- Ograniczenia *Track To* należy użyć do obiektów, które mogą się obracać w dwóch kierunkach — na przykład cięgien (str. 439);
- Zamykanie pokryw podwozia można uzyskać za pomocą obracanego „ramienia”, zakończonego obiektem — „celem”. Ten „cel” jest śledzony przez osie **Y** zawiasów (bo mają przypisane ograniczenia *Locked Track* — str. 441). Trajektoria tego „celu” może nie być okręgiem (str. 442);
- Ograniczenie *Stretch To* można użyć do „przyłączenia” do mechanizmu popychaczy. Unikniemy w ten sposób „odrywania” ich końcówek od okuć, występującego w niektórych pośrednich pozycjach mechanizmu podwozia (str. 443);

### 7.3 Podwozie główne

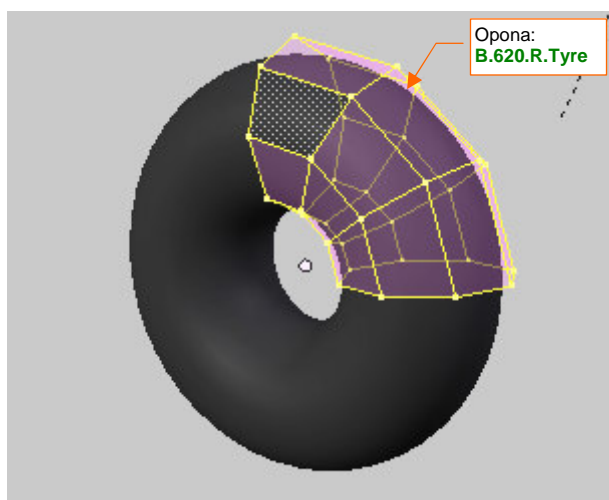
Jak przekonałeś się w poprzednich sekcjach, podwozie składa się z dziesiątek drobnych elementów. Zazwyczaj są one zbyt małe, by zostać prawidłowo nakreślone na planach modelarskich. O ile autor nie załączył jakiegoś rysunku szczegółowego, podczas wykonywania podwozia jesteś skazany na analizę dostępnych zdjęć. W przypadku P-40 jesteśmy w tej szczęśliwej sytuacji, że wśród udostępnionych w Internecie dużej liczby fotografii, można znaleźć także oryginalny rysunek złożeniowy podwozia głównego (widok z przodu i z boku). Pozostaje wkomponować go w rzuty samolotu (Rysunek 7.3.1):



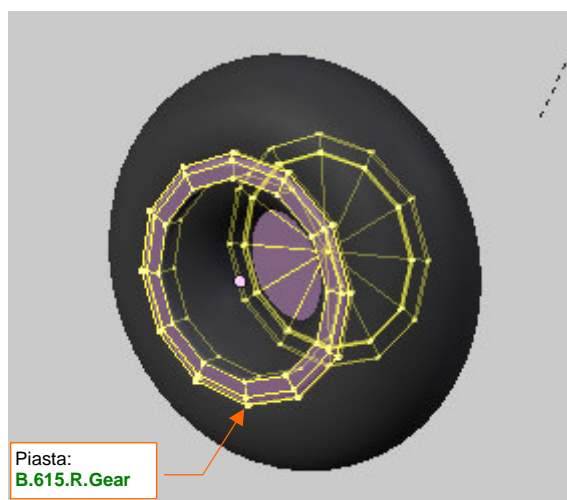
Rysunek 7.3.1 Fabryczne rysunki zespołu koła głównego

Oryginalny rysunek fabryczny podzieliłem na rzut z przodu i rzut z boku. Dopasowywanie zacząłem od rzutu z przodu, gdyż zawiera zarys dolnej krawędzi przedniego dźwigara. (Było do niej przymocowane okucie osi obrotu goleni). Na siatce skrzydła modelu zidentyfikowałem podłużnicę, która odpowiada tej krawędzi, i przenieśliem jej zarys na plany (Rysunek 7.3.1a). W oparciu o tę linię, można było poprawnie „zawiesić” podwozie pod skrzydłem. Potem dla tej samej współrzędnej **Z** i skali ustawiłem rzut z boku (Rysunek 7.3.1b). Aby nie wierzyć ślepo planom, położenie goleni wzdłuż osi samolotu (ich współrzędną **Y**) ustaliłem na podstawie zdjęć.

Modelowanie podwozia zaczynam od opony. Wykonałem ją tą samą techniką, co kółko ogonowe — jako „ćwiartkę” torusa, odbitą modyfikatorem *Mirror* względem dwóch osi (Rysunek 7.3.2).



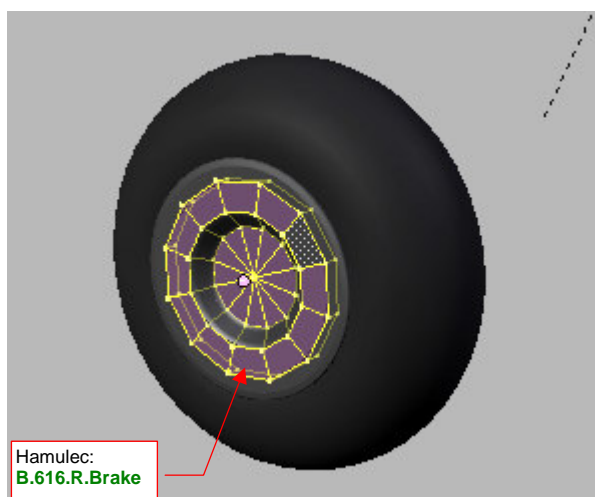
Rysunek 7.3.2 Siatka opony



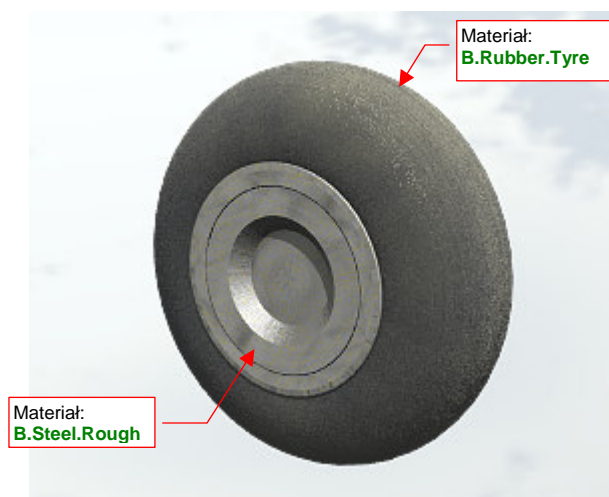
Rysunek 7.3.3 Siatka piasty koła

(Ze zdjęć wynika, że opony wczesnych wersji P-40 zupełnie nie miały bieżnika. W razie czego — bieżnik można zamodelować lub odwzorować za pomocą tekstury nierówności). Podczas wykonywania piasty koła nie skorzystałem z modyfikatora (Rysunek 7.3.3). Tarcze piast były często na P-40 malowane w fantazyjne wzory, więc jako jedyny element podwozia jej siatka zostanie rozwinięta w płaszczyźnie UV.

W bardzo podobny sposób wykonałem tarczę hamulca (Rysunek 7.3.4). Choć wygląda jak fragment piasty, to, w odróżnieniu od niej, była „na sztywno” przymocowana do osi koła, więc musi być odrębnym obiektem.



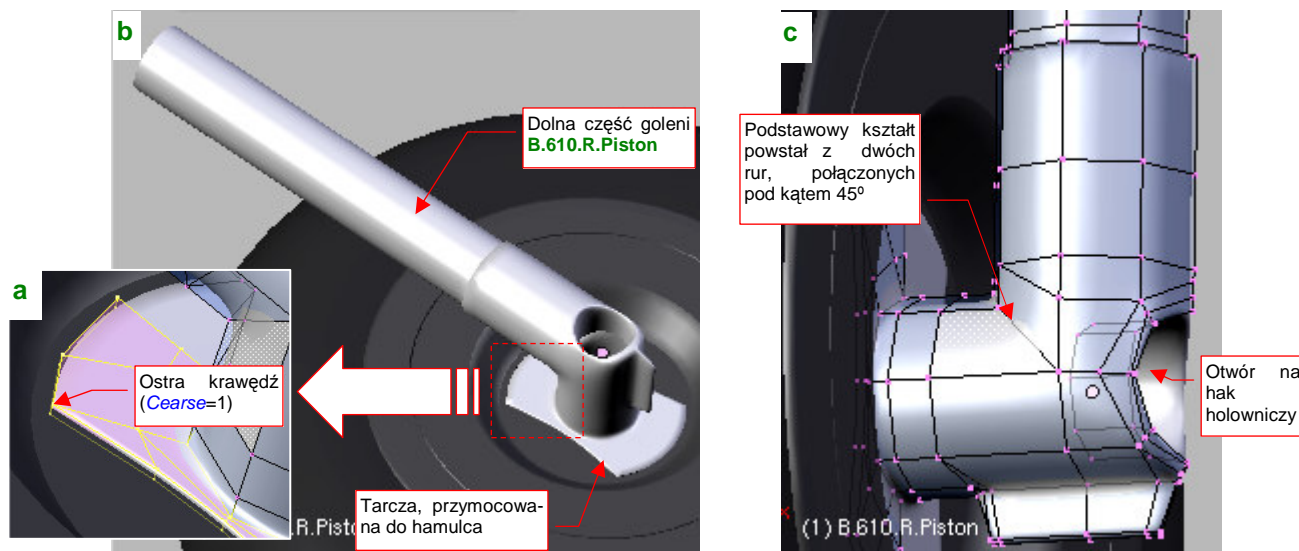
Rysunek 7.3.4 Siatka tarczy hamulcowej



Rysunek 7.3.5 Koło główne — obiekty i ich materiały

Nadałem tej oponie nazwę **B.620.R.Tyre**, piaście — **B.615.R.Gear**, a tarczy hamulca — **B.616.R.Brake**. Zbudowałem z nich prostą hierarchię, przypisując oponę do piasty, a piastę do hamulca. Zaraz po skończeniu modelowania warto także przypisać poszczególnym siatkom odpowiednie materiały, by już do nich nie wracać. Przy okazji pracy nad kółkiem ogonowym „dorobiliśmy” się już odpowiedniego zestawu wykończeń. Przydzieliłem oponie koła głównego ten sam materiał, jaki wykorzystaliśmy na oponę kółka ogonowego: **B.Rubber.Tyre**. Pozostałym elementom przypisałem typową powłokę metalu, czyli **B.Steel.Rough** (Rysunek 7.3.5).

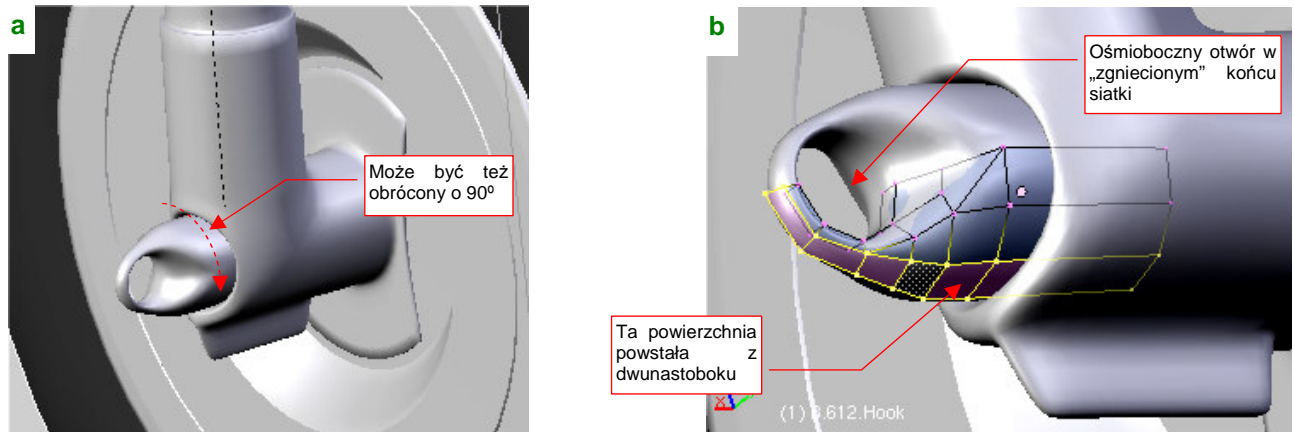
Teraz wykonamy dolną część goleni, pełniącą jednocześnie rolę tłoka amortyzatora (Rysunek 7.3.6):



Rysunek 7.3.6 Dolna część goleni — tłok amortyzatora

Podstawowy kształt tej siatki powstał ze złożenia pod kątem  $45^\circ$  dwóch rur (Rysunek 7.3.6c). Następnie wyznaczyłem na tylnej ścianie okrągłą krawędź otworu na zaczep (hak) holowniczy. (Zrobiłem to za pomocą skryptu [Cross Section](#) — por. str. 803). W oparciu o tę krawędź wkomponowałem w siatkę goleni odpowiedni otwór. Z przeciwnej strony dodałem do siatki okrągłą tarczę, do której w prawdziwym samolocie był przykręcony zespół hamulca. Przy odpowiednim przesunięciu i zagęszczeniu wierzchołków, udało mi się przekształcić ten ośmiokąt w kształt przypominający klepsydrę (Rysunek 7.3.6a). Oczywiście, uzyskanie ostrych krawędzi było tylko możliwe poprzez zwiększenie ich wartości **Cearse** do 1.0. Zamodelowałem tylko jedną stronę siatki — reszta jest wygenerowana przez modyfikator **Mirror**. Nadałem tej części goleni nazwę **B.610.R.Piston** i ustaliłem materiał na **B.Steel.Rough**. W hierarchii modelu ta goleń jest „rodzicem” tarczy hamulca.

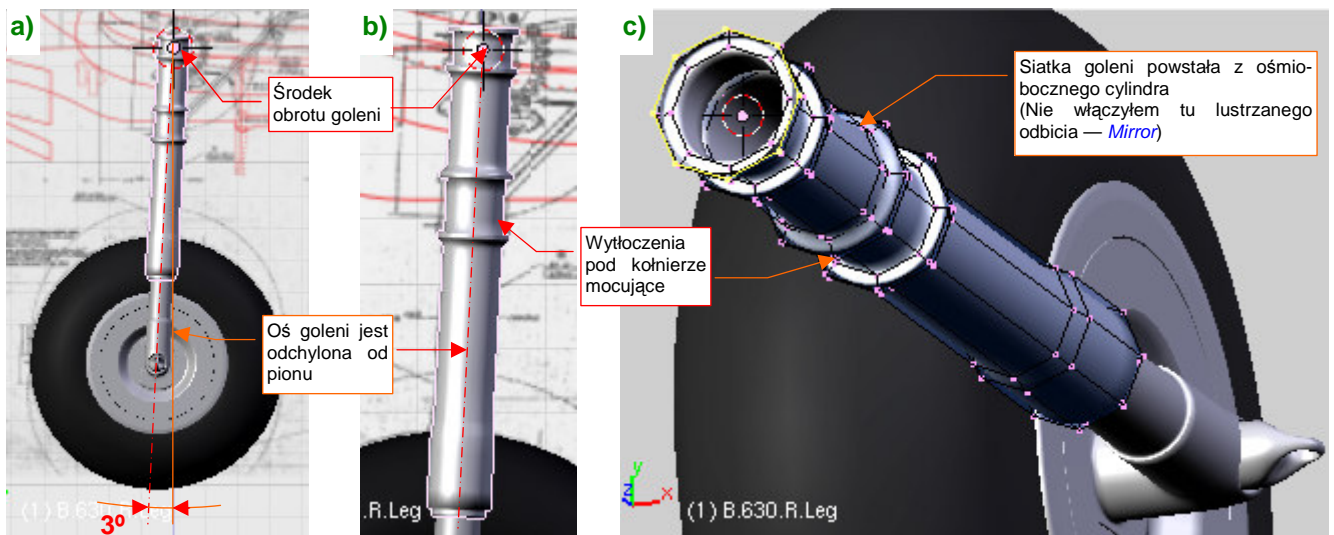
Kolejnym fragmentem podwozia jest zaczep (hak) holowniczy (Rysunek 7.3.7):



Rysunek 7.3.7 Hak do holowania

Według rysunków fabrycznych, hak powinien być ustawiony pionowo (Rysunek 7.3.7a), choć na zdjęciach odrestaurowanych samolotów najczęściej jest obrócony o  $90^\circ$  (w tym położeniu najprawdopodobniej łatwiej jest przymocować do każdego z kół końcówkę jakiegoś pomocniczego „zaprzęgu” holowniczego). Siatka tej części to dwunastoboczny cylinder, któremu „zgniotłem” końcówkę, wkomponowując w nią otwór, oparty na ośmioboku (Rysunek 7.3.7b). Aby się nie przemęczać, modelowałem tylko „ćwiartkę” tej powierzchni — reszta to rezultat użycia modyfikatora *Mirror*, generującego odbicie względem dwóch osi. Nadałem tej części nazwę **B.612.Hook**, i umieściłem w hierarchii obiektów jako „potomka” goleni (**B.610.R.Piston**). Tak jak pozostałym elementom podwozia, przypisałem do siatki **B.612.Hook** materiał **B.Steel.Rough**.

Jak się powiedziało „A”, trzeba powiedzieć i „B”, więc skoro mamy dolną część goleni (tłok amortyzatora), dodajmy zaraz górną, zawierającą jego cylinder (Rysunek 7.3.8):

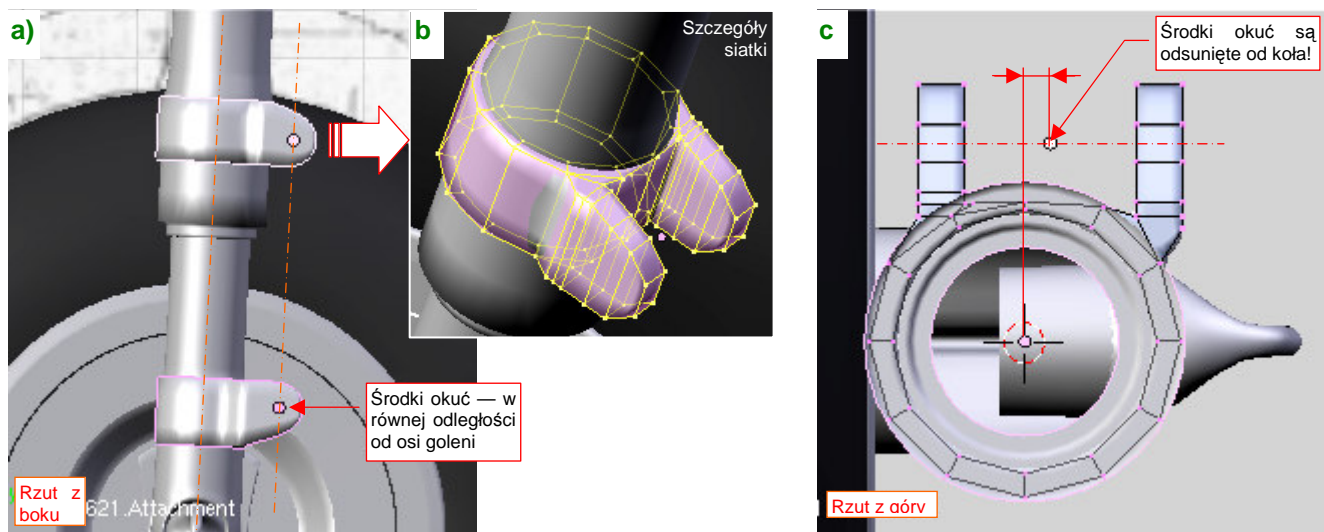


Rysunek 7.3.8 Druga część goleni — cylinder amortyzatora

Cylinder amortyzatora stanowił integralny fragment górnej części goleni podwozia. Siatka tego elementu powstała z ośmioboku (Rysunek 7.3.8c). To stosunkowo prosty kształt, jak z tokarki. Nie redukowałem tu liczby oryginalnych ścian za pomocą modyfikatora *Mirror* — bo z jednej strony ta siatka będzie miała „wypustkę” (por. str. 453). Goleń podwozia P-40 ma lekko pochyloną oś. Podczas pracy będziesz wielokrotnie ustawiał ją na chwilę do pionu, więc warto ustalić jej pochYLENIE na jakiś okrągły kąt, by łatwo było w to położenie wrócić. Po sprawdzeniu planów i zdjęć, zdecydowałem się na pochYLENIE osi goleni dokładnie o  $3^\circ$  (Rysunek 7.3.8a).

Goleń podwozia P-40 podczas skłAdnia obracała się wokół swojej osi o ponad  $90^\circ$ . Utrzymują ją obejmY osi obrotu i popychaczy. Stąd na powierzchni **B.630.R.Leg** są przygotowane małe wytłoczenia dla tych obejm.

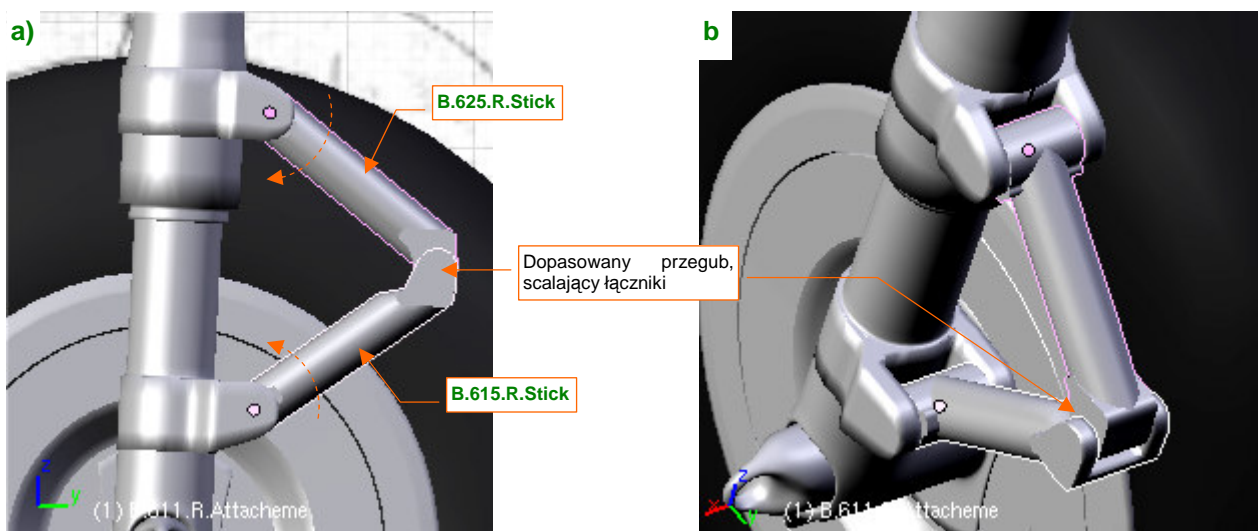
Aby zapobiec przypadkowemu obrotowi, górna i dolna część goleni są w samolotach łączone za pomocą dwóch przegubowych łączników (Rysunek 7.3.10). Modelowanie tego zespołu zaczniemy od ich okuc (Rysunek 7.3.9):



Rysunek 7.3.9 Okucia łączników

Okucia łączników w P-40 wyglądają jak dwa „paski”, wygięte wokół goleni (Rysunek 7.3.9a). Wydaje mi się, że takie rozwiązanie znajdziesz na większości samolotów z okresu II Wojny Światowej. Technologicznie nie opłacało się wykonać „w jednym kawałku” goleni razem z okuciami. Rysunek 7.3.9b pokazuje szczegóły siatki okuc — to w zasadzie cylinder z dołączonymi „uszami” na oś obrotu łącznika. Świadomie zrezygnowałem tu z modelowania otworu osi (por. np. 427, Rysunek 7.1.18) — bo zostanie i tak zostanie zupełnie zasłonięty przez łeb śruby lub nakrętkę (na przeciwnej ściance). Zwróć uwagę, że „uszy” okuc są odsunięte w bok od koła (Rysunek 7.3.9c). Tak wynika ze zdjęć (gdyby płaszczyzna symetrii łączników leżała na osi goleni, wówczas nakrętka śruby zawadzałaby o oponę). Środki okuc umieściłem w osiach obrotu łączników (są równoległe, ale odsunięte od goleni — por. Rysunek 7.3.9a). Takie położenie będzie w przyszłości bardzo przydatne do zbudowania poprawnego „mechanizmu” dla tego zespołu. Górnemu okuciu nadałem nazwę **B.621.R.Attachment**, a dolnemu — **B.611.Attachment**.

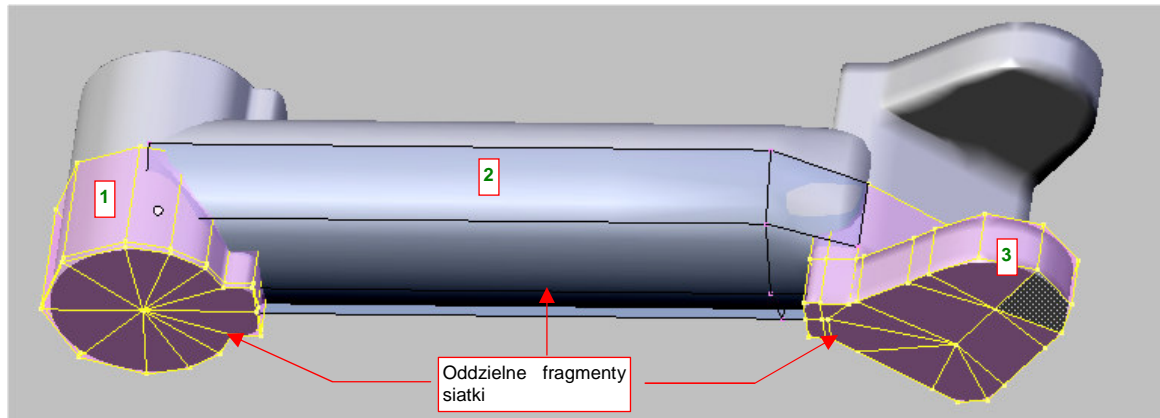
Gdy mamy już w czym „osadzić” łączniki, można je wykonać (Rysunek 7.3.10):



Rysunek 7.3.10 Łączniki tłoka i cylindra amortyzatora

Środki łączników znajdują się w tym samym punkcie, co środki ich okuc (por. Rysunek 7.3.10a i Rysunek 7.3.9a). Łączniki nie są identyczne — dolny jest zakończony „widelcem”, który zachodzi na końcówkę górnego (Rysunek 7.3.10b).

Sam łącznik wygląda na solidną, krótką sztangę. Jest na tyle drobny, że tworząc jego siatkę zrezygnowałem z pracochłonnego łączenia poszczególnych zespołów (Rysunek 7.3.11):

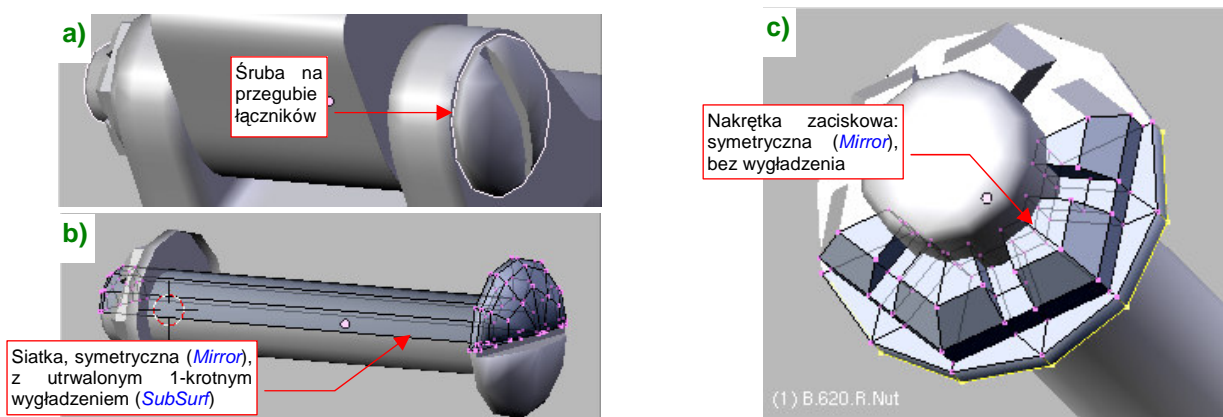


Rysunek 7.3.11 Szczegóły siatki łącznika amortyzatora

Siatka łącznika powstała ze złożenia trzech oddzielnych powłok: cylindra wokół osi okucia (1), właściwego ramienia (2), oraz widelca (3) (w górnym łączniku ma inny kształt). Mimo ostrych krawędzi przenikających się powierzchni, element w *Object Mode* i na renderze wygląda to całkiem normalnie. Trzeba tylko dokładnie dosunąć do siebie wszystkie trzy powłoki, by nigdzie nie było widać jakichś szczelin.

Wreszcie doszliśmy do śrub (Rysunek 7.3.12a). Tak, tak, zdecydowałem się zamodelować co większe egzemplarze, choć sam wcześniej ostrzegałem przed popadaniem w przesadę. (I zamiast modelować np. śruby mocujące owiewki połączeń skrzydła i kadłuba, odwzorowałem je za pomocą tekstury nierówności — por. str. 312). Tym razem jednak robię to z całą premedytacją. Po pierwsze — śruby, wykorzystane do połączenia elementów goleni miały większy rozmiar od tych, wykorzystywanych do mocowania blach płatowca. Po drugie — do modelowania podwozia stosujemy tylko tekstury proceduralne, więc nie ma gdzie narysować odpowiednich łbów i nakrętek. Po trzecie — aby poprawnie „zmechanizować” ruch łączników towarzyszący ugięciu amortyzatora, potrzebujemy oddzielnego obiektu w każdej z trzech osi obrotu tego zespołu. Po co wstawiać w to miejsce jakieś „puste” (*Empty*) twory, skoro możemy w naturalny sposób skorzystać ze śrub, umieszczonych tam przez konstruktora?

Siatkę śruby wygenerowałem z połówki cylindra, z włączonym modyfikatorem *Mirror* (Rysunek 7.3.12b):

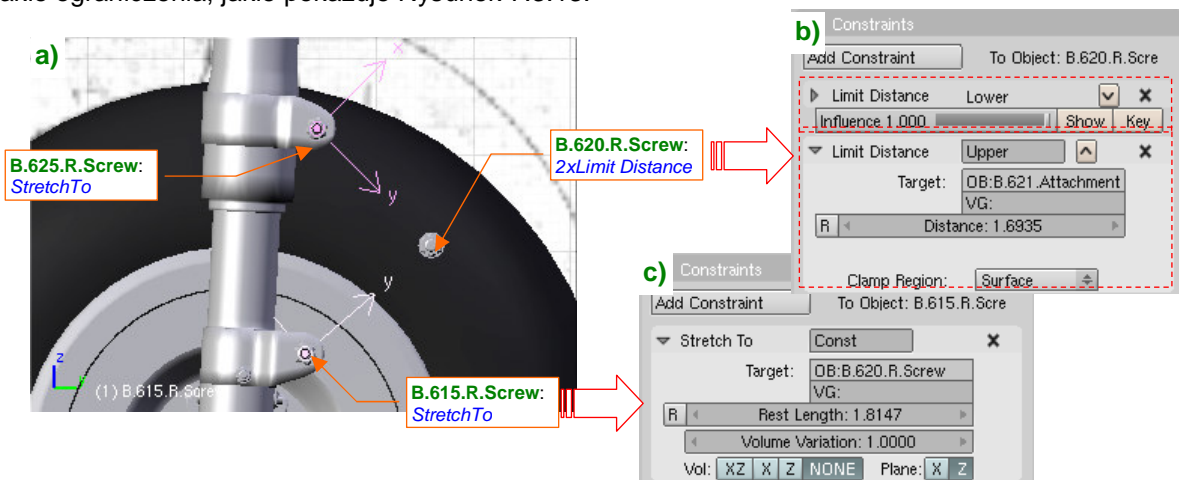


Rysunek 7.3.12 Śruba i nakrętka

Kulisty łeb śruby wytłoczyłem z końcówki walca. Po włączeniu umiarkowanego wygładzenia (modyfikator *Subsurf* z *Level* = 1) utrwaliłem je (*Apply*). Zrobiłem to po to, by wygodnie naciąć we łbie szczelinę. Śruby wykorzystamy w wielu miejscach podwozia, więc przygotowałem kilka alternatywnych siatek, różniących się długością gwintu (**B.Screw.Flat.008**, **B.Screw.Flat.010**, **B.Screw.Flat.011** — końcówki oznaczają długość, w 1/10 jednostki Blendera). Zastosujemy je w wielu klonach. Nakrętka zaciskowa („bezpieczna” — **B.Nut.Safe**) ma siatkę na bazie dwunastoboku, bez żadnych wygładzeń (Rysunek 7.3.12c).

Nie wiem, czy w oryginalnym P-40 stosowano takie zaciskowe nakrętki, jakie pokazuje Rysunek 7.3.12c). Zamodelowałem ją na podstawie zdjęć jednego z odrestaurowanych P-40E.

Umieściłem śruby w osiach ramion łącznika. Same ramiona można na chwilę ukryć (**H** — *Hide*). Przypisałem im takie ograniczenia, jakie pokazuje Rysunek 7.3.13:

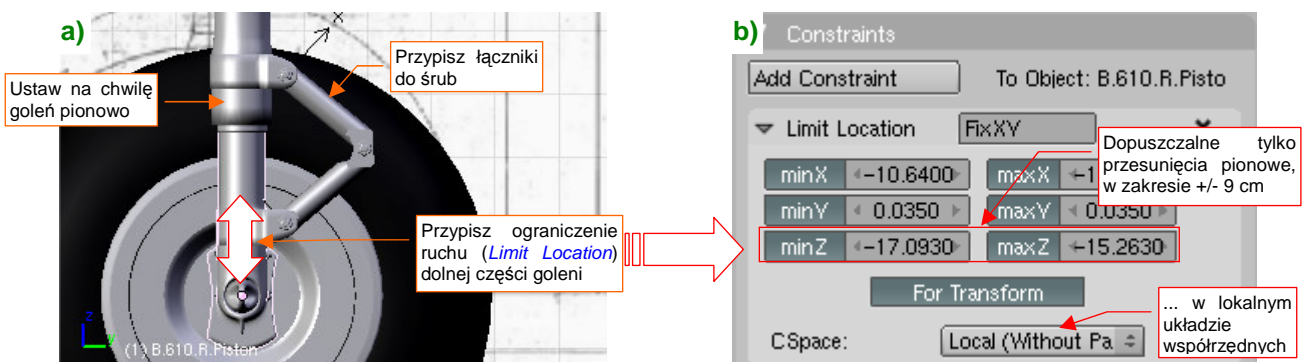


Rysunek 7.3.13 Budowa mechanizmu przegubowego połączenia goleni.

Śrubie umieszczonej w osi przegubu nadałem nazwę **B.620.R.Screw** (Rysunek 7.3.13a). Przypisałem jej dwa ograniczenia *Limit Distance* (Rysunek 7.3.13b). Każde z nich wymusza utrzymanie przez ten obiekt stałej odległości (*Clamp Region* = *Surface*) od środków okuc (leżą w tym samym miejscu, co osie obrótu ramion). Śruby umieszczone w tych osiach — **B.625.R.Screw** i **B.615.R.Screw** — przypisałem (relacją *Parent*) do okuc. Nadałem im ograniczenia *Stretch To*, by ich osie *Y* były zawsze skierowane na oś przegubu. Potem możesz z powrotem „odstąpić” (**Alt-H**) schowane przed chwilą ramiona. Przypisz każdą z nich (relacją *Parent*) do odpowiedniej śruby (**B.625.R.Screw** dla górnego ramienia, **B.615.R.Screw** dla dolnego). Powinieneś uzyskać rezultat jaki pokazuje Rysunek 7.3.14a).

- Szczegółowy opis budowy tego mechanizmu znajdziesz w sekcji dotyczącej ograniczenia *Limit Distance*, na str. 835. Wykorzystałem tam go jako przykład.

Aby wymusić ruch dolnej części amortyzatora tylko w założonych granicach (max. skok tłoka — +/- 9 cm), przypiszemy mu ograniczenie *Limit Location* (Rysunek 7.3.14):

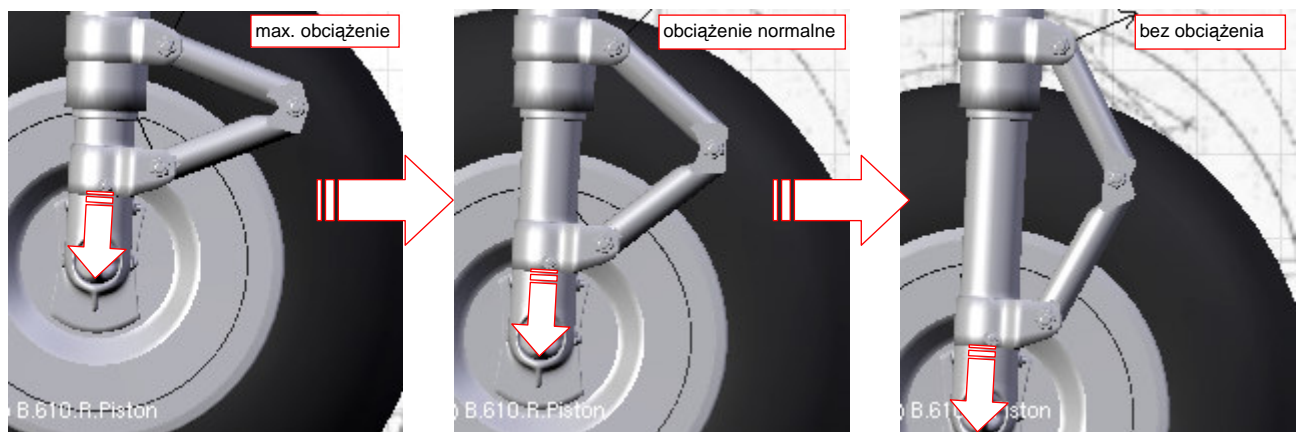


Rysunek 7.3.14 Ograniczenie ruchu amortyzatora.

Uważaj: przed przypisaniem tłokowi ograniczenia *Limit Location*, ustaw na chwilę całą goleń zupełnie pionowo (Rysunek 7.3.14a). (To dlatego poprzednio podkreślałem, że powinna być pochylona o jakąś „okrągłą” wartość kąta, by łatwo było ją na chwilę przywrócić do pionu — por. str. 447). Tylko w takim położeniu Blender poprawnie „zrozumie”, co masz na myśli pozostawiając obiektowi swobodę ruchu tylko wzdłuż osi *Z* (Rysunek 7.3.14b). Potem możesz z powrotem pochylić całą goleń — dzięki przełączeniu *Cspace* na *Local*, kierunki działania ograniczeń obróca się wraz z nią.



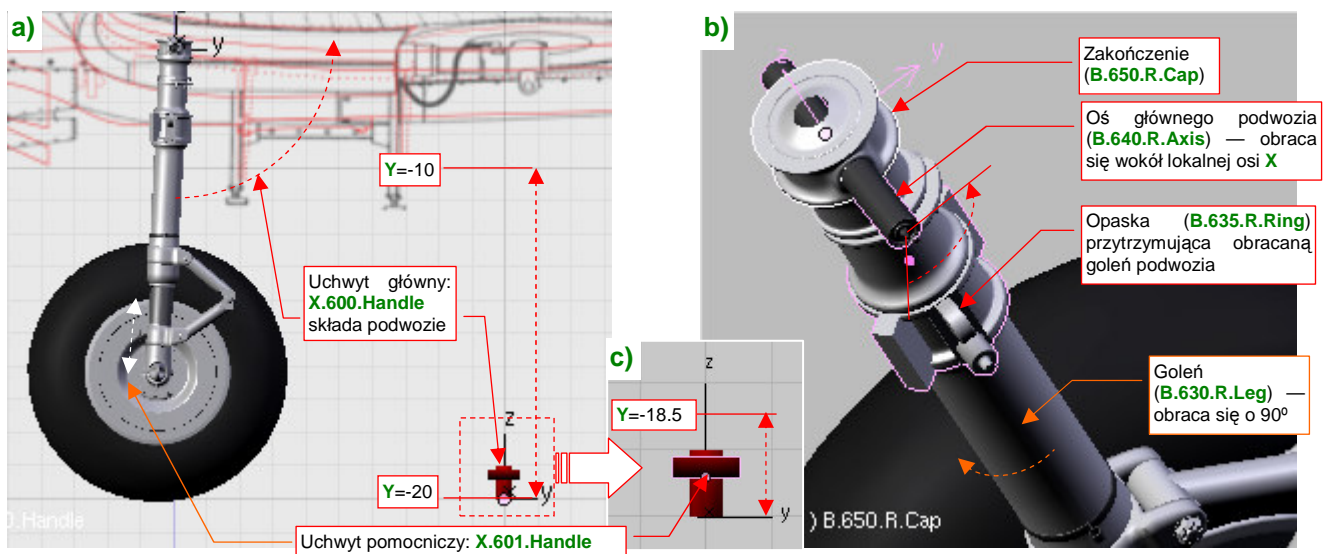
Rysunek 7.3.15 przedstawia działanie mechanizmu, który zbudowaliśmy. Wraz z ruchem tłoka ramiona łączników poprawnie się składają i rozkładają:



Rysunek 7.3.15 Działanie mechanizmu przegubowego połączenia goleni.

Zapewne, ten sam efekt można było uzyskać za pomocą armatury, złożonej z dwóch kości (por. str. 815). Jednak w poprzedniej sekcji zdecydowałem się wyrugować armatury z mechanizmu kółka ogonowego. Aby zachować jednorodność rozwiązań w modelu, muszę teraz konsekwentnie wykonać każdy mechanizm wyłącznie za pomocą ograniczeń. Zresztą — to nie jest aż takie trudne, jak mogłoby się wydawać.

Czas pomyśleć o dalszej „mechanizacji” naszego podwozia. Zaczniemy od przygotowania dwóch uchwytów, działających w sposób podobny do uchwytów kółka ogonowego (Rysunek 7.3.16a):

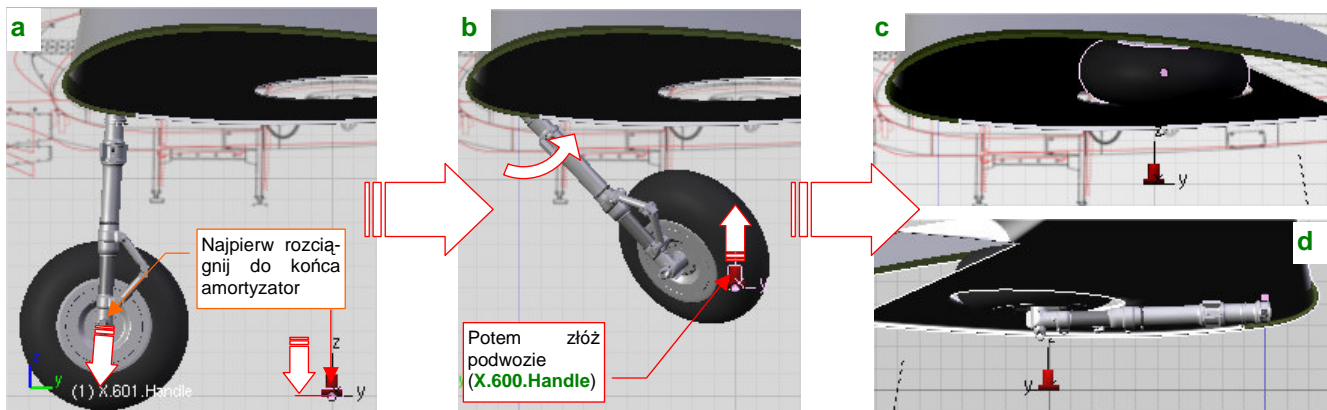


Rysunek 7.3.16 Podstawowe elementy animacji goleni

Podstawowym uchwytem, odpowiedzialnym za złożenie podwozia, jest pomocniczy obiekt **X.600.Handle**. Jego ruch ograniczyłem do osi **Y**, od -10 (podwozie schowane) do -20 (zupełnie wysunięte). **X.600.Handle** jest „rodzicem” pomocniczego uchwytu **X.601.Handle** (Rysunek 7.3.16c). To uchwyt sterujący ugięciem amortyzatorów. Ograniczyłem jego ruch do 1.5 jednostki Blendera (15 cm), by dokładnie odpowiadał zakresowi ugięcia podwozia głównego. (Z rysunków fabrycznych wynika, że dla normalnej masy startowej ugięcie wynosiło 6cm. Gdy zakresy ruchu amortyzatora i sterującego uchwytu są takie same, ustawienie go w odpowiedniej pozycji nie nastęrcza problemów).

Goleń podwozia P-40 nie tylko obracała się do tyłu, ale także wokół swojej osi. Aby dobrze odwzorować ten ruch, odtworzyłem elementy oryginalnego podwozia (Rysunek 7.3.16b). Dodałem oś — **B.640.R.Axis** — obracaną za pomocą uchwytu **X.600.Handle** (poprzez ograniczenie *Transform*). Bezpośrednio do osi przypisane są trzy elementy: opaska (**B.635.R.Ring**) i „czepiec” (**B.650.R.Cap**), oraz sama goleń (**B.630.R.Leg**). Podczas składania podwozia goleń dodatkowo obraca się wokół swojej osi (w wyniku ograniczenia *Transform*) o 90°.

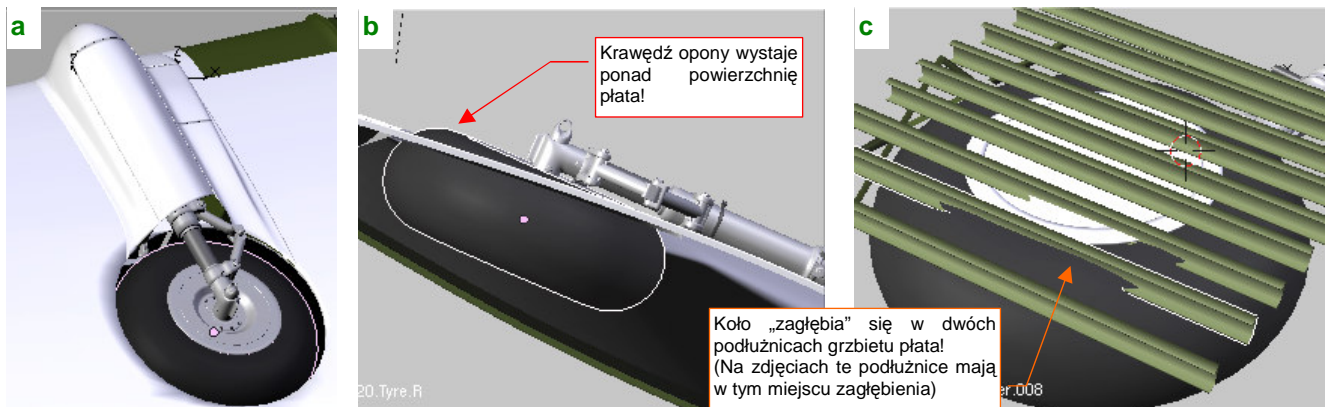
A jak to wygląda w działaniu? Składanie podwozia zacznij od rozciągnięcia amortyzatora (za pomocą uchwytu **X.610.Handle**) do pozycji swobodnej (Rysunek 7.3.17a):



Rysunek 7.3.17 Składanie podwozia głównego

(Inaczej koło nie trafi w swój otwór!). Następnie złap za główny uchwyt (**X.600.Handle**) i zacznij go przesuwając do góry. Spowoduje to złożenie goleni, połączone z jednoczesnym obrotem (Rysunek 7.3.17b) — tak jak w prawdziwym P-40. Zakres obrotu **B.640.R.Axis** wokół lokalnej osi **X** — od  $-3^\circ$  do  $+88^\circ$ . Zakres obrotu goleni **B.630.R.Leg** wokół lokalnej osi **Z** — od  $0^\circ$  do  $+96^\circ$  (trochę więcej niż  $90^\circ$ , by płaszczyzny kół leżały na powierzchni płata o wzniosie  $+6^\circ$ ). Rysunek 7.3.17c, d) pokazuje podwozie główne w pozycji złożonej.

Dopasowanie odpowiednich kątów złożenia wymaga paru iteracji, podczas których należy uważnie sprawdzać położenie koła względem pozycji płata i ścian komory. Te, które podałem, dobrałem na podstawie rysunków fabrycznych. Pozwalają złożyć podwozie tak, jak to pokazuje Rysunek 7.3.18a):



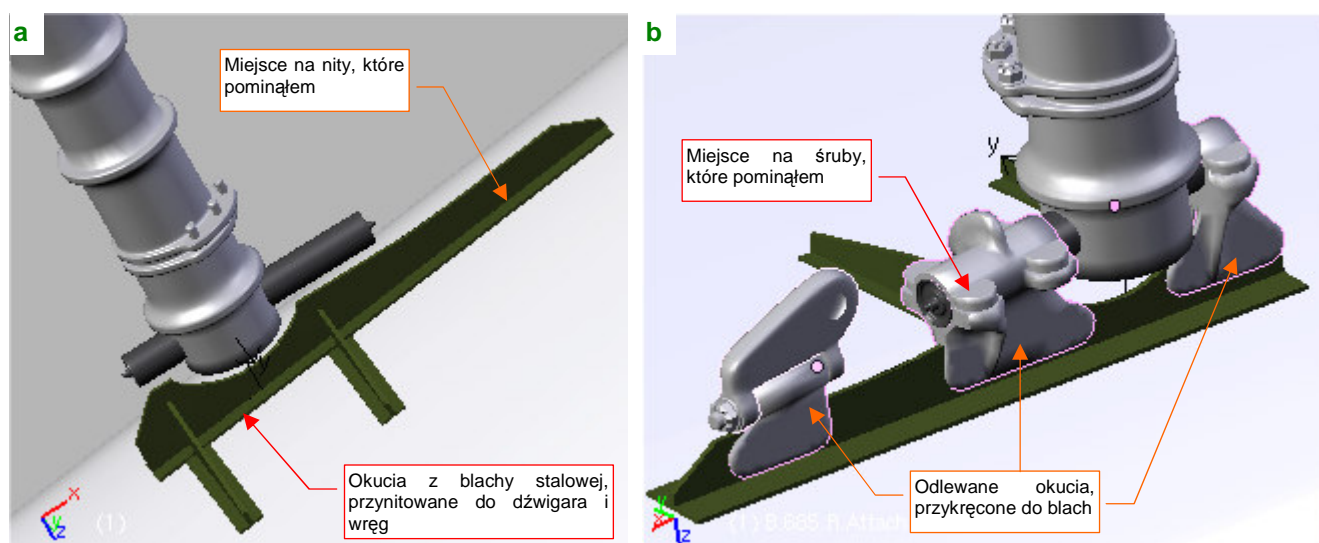
Rysunek 7.3.18 Dopasowanie kół do komory podwozia

Wydaje się być całkiem poprawne — tak samo złożone podwozie wygląda na dostępnych zdjęciach P-40, wykonanych podczas lotu. Przy okazji dopasowywania wyszły na jaw ciekawe szczegóły:

- Tylna krawędź opony wystaje z dolnej powierzchni skrzydła (Rysunek 7.3.18b)! Sprawdzałem trzy razy rozmiar opony, kąt pochylenia złożonej goleni, i doszedłem do wniosku, że tak było naprawdę<sup>1</sup>. Jak pokazuje Rysunek 7.3.18a), w większości ujęć zupełnie tego nie widać, gdyż otwór komory jest nieco większy od koła. Dzięki temu czarna, matowa, wystająca opona nie rzuca się w oczy;
- Wewnętrzna powierzchnia koła zagłębia się w dwóch tylnych podłużnicach górnego pokrycia płata (Rysunek 7.3.18c)! Dzięki temu zrozumiałem, dlaczego te dwie podłużnice na zdjęciach mają, wzdłuż komory podwozia, wyraźne wgłębienia. Nie zamodelowałem ich, bo mało kto w to miejsce modelu będzie zaglądał. Oczywiście, jeżeli chcesz wykonać dokładniejszy model, nie zapomnij o tym szczególe;

<sup>1</sup> P-36 Hawk i nawet prototyp XP-40 miały mniejsze koła, które dokładnie mieściły się w płacie. To większe podwozie wprowadzono podczas przygotowywania produkcji seryjnej. Z tego powodu został wygięty tylny dźwigar (por. str. 138, Rysunek 4.7.2). Przy tak poważnej zmianie parę centymetrów gumy pod skrzydłem nic nie znaczyło! Prawdopodobnie to z tego powodu zrezygnowano z pokryw kół, które miał P-36.

Uruchomiliśmy już podwozie, ale czas je wzbogacić o resztę „żelastwa”, które widać na zdjęciach. Zaczni od przygotowania okuć, które mocowały oś do skrzydła (Rysunek 7.3.19a):

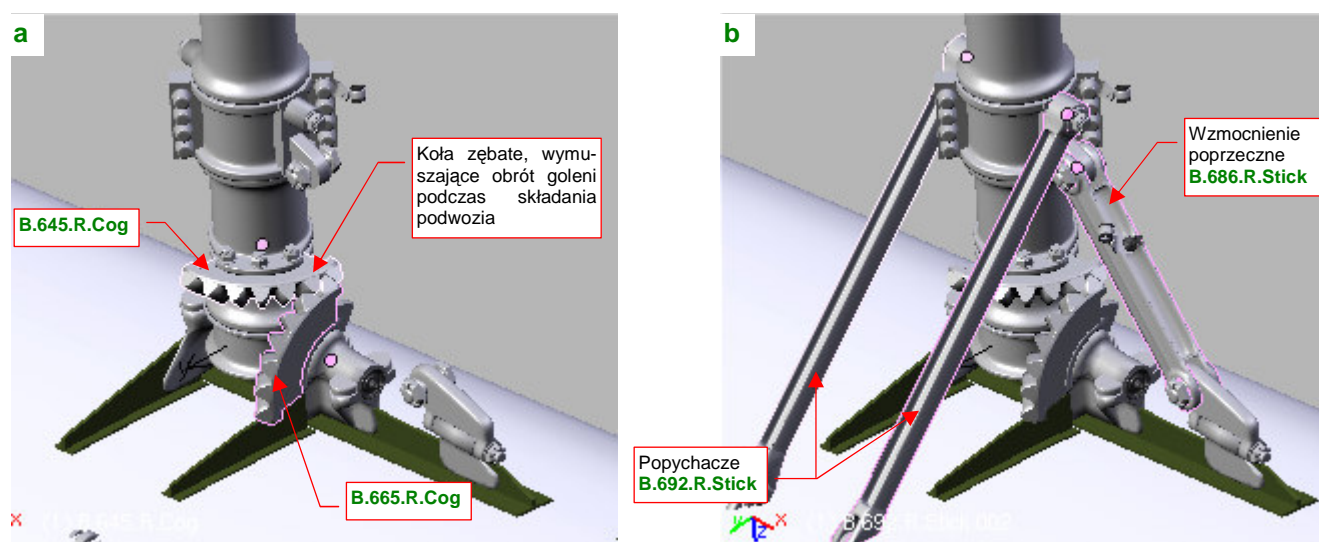


Rysunek 7.3.19 Okucia osi podwozia

Na zdjęciach i rysunkach konstrukcyjnych P-40 widać blachy (lub odkuwki) o profilu w kształcie „T”, przynitowane do dźwigara i wręg. Zamodelowałem je jako jeden obiekt (**B.660.R.Frame**), starannie dopasowując do powierzchni skrzydła. Gdybyś chciał robić jakiś superdokładny model, należałoby odwzorować — np. za pomocą mapy nierówności — wypukłe nity, którymi przymocowano półki tych elementów do płata. Ja jednak przypisałem tym wzmocnieniom standardowy materiał **B.Inner.Details**, bez nitów. I tak nie będzie ich prawie widać.

Do blach okuć były przykręcone (za pomocą wielu śrub) kolejne elementy, mocujące oś podwozia (Rysunek 7.3.19b). Mówiąc szczerze, mimo rysunków i paru zdjęć, jakie miałem do dyspozycji, trudno było zdeterminować ostateczny kształt tych części. Kilka razy musiałem je szkicować na kartce papieru, by uświadomić sobie ich wszystkie zakamarki. I w tym przypadku pominąłem kilkanaście śrub, którymi były przykręcone do blach okuć — ich także prawie nie widać.

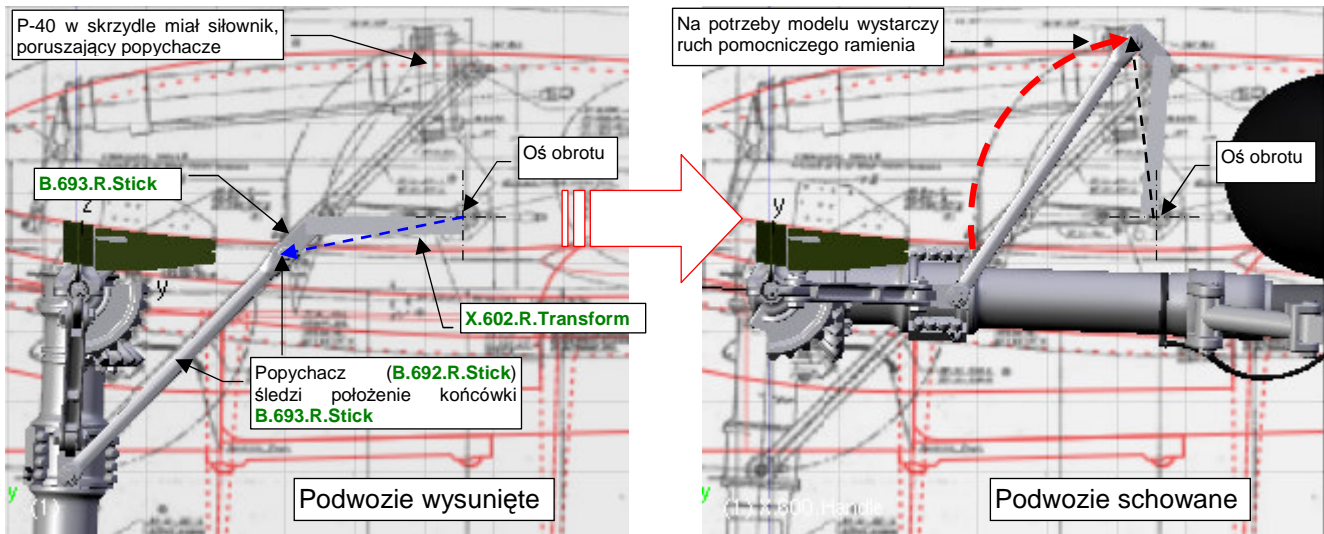
Następnie dodałem dwie zębatki, które wymuszały obrót goleni podczas składania podwozia (Rysunek 7.3.20a):



Rysunek 7.3.20 Dalsze szczegóły podwozia

Zębatka **B.645.R.Cog** jest przypisana do obracającej się goleni **B.630.R.Leg**. (W tym momencie przydało się zamodelowanie goleni jako pełnej siatki: można było z niej łatwo wytłoczyć niesymetryczną wypustkę do umocowania zębatki). Zębatkę **B.665.R.Cog** przypisałem do nieruchomych okuć osi. Potem dodałem pozostałe elementy: popychacze (**B.692.R.Stick**) i wzmocnienie poprzeczne (**B.686.R.Stick**) (Rysunek 7.3.20b). Każde z nich jest przypisane (relacją **Parent**) do opaski **B.635.R.Ring**, by składały się wraz z resztą podwozia.

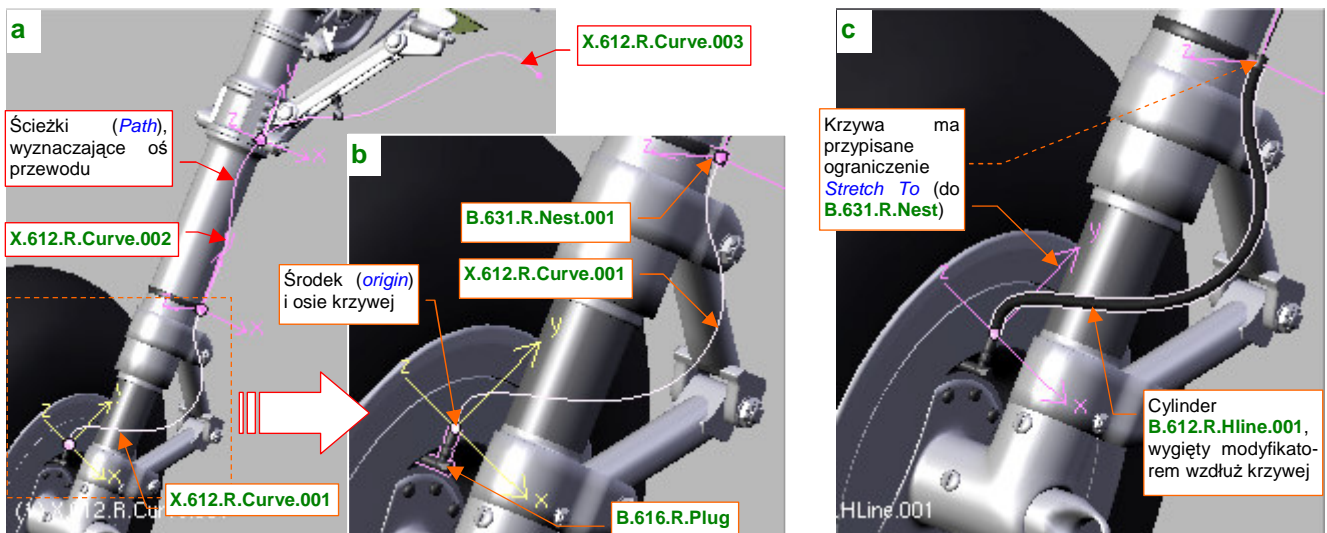
Aby popychacze wykonywały ruch przypominający rzeczywisty, dodałem na ich krańcach końcówki (**R.693.R.Stick**). Są one przypisane do zupełnie niewidocznej ramki, która je obraca (**X.602.R.Transform**) (Rysunek 7.3.21):



Rysunek 7.3.21 Mechanizm popychaczy

Ramka **X.602.R.Transform** ma przypisane ograniczenie **Transform**. Dzięki temu podczas składania podwozia wykonuje obrót o taki sam kąt, jak ramka popychaczy. Do ramki są przypisane (relacją **Parent**) końcówki **R.693.R.Stick**, więc obracają się wraz z nią. Popychacze **B.692.R.Stick** mają ustawione ograniczenia **Locked Track** na swoje końcówki, więc podczas chowania podwozia poruszają się po trajektorii bardzo zbliżonej do rzeczywistej. Mechanizmu, ukrytego wewnątrz skrzydła, nie modelowałem. I tak jest zupełnie niewidoczny.

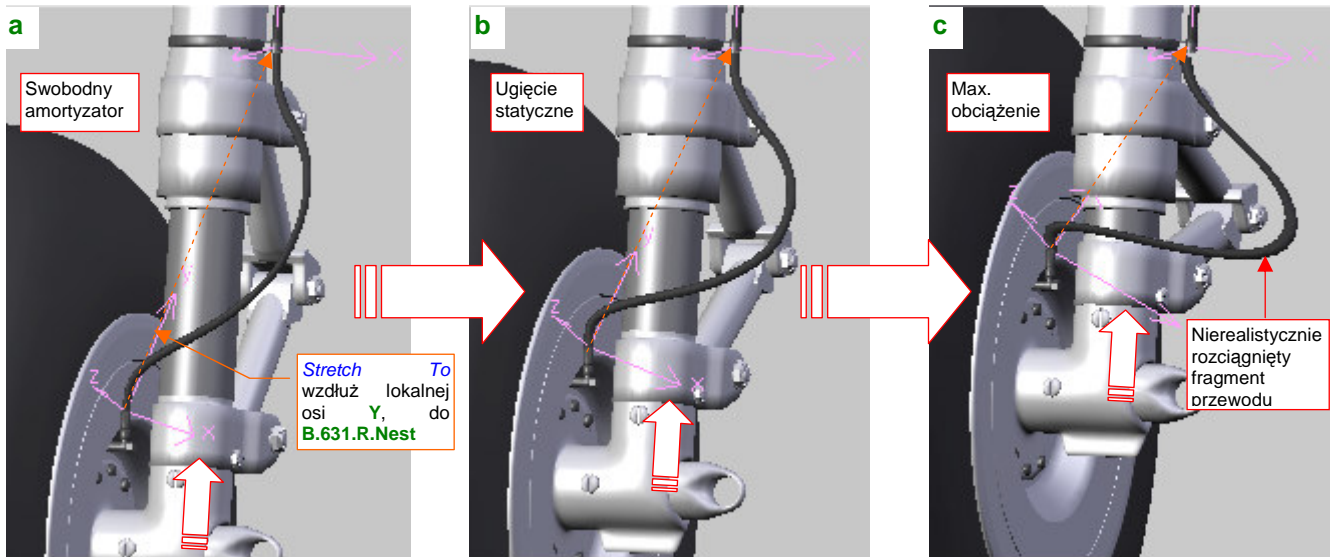
Wzdłuż goleni P-40 (i większości samolotów) biegnie elastyczny przewód z płynem hamulcowym. Ten detale zamodelowałem za pomocą krzywej (**Curve**: szczegóły — str. 842), wokół której „owinałem” rurkę (cylinder) przewodu (za pomocą polecenia **Parent→Curve Deform**, opisanego na str. 913). Gdyby chodziło o statyczny model, faktycznie to by wystarczyło. Nasza goleń potrafi się jednak obrócić, i może się w niej ugiąć amortyzator. To zmusiło mnie do zamodelowania przewodu za pomocą trzech oddzielnych krzywych (Rysunek 7.3.22a):



Rysunek 7.3.22 Odwzorowanie przewodu hamulcowego

Krzywe — **X.612.R.Curve.001** .. **003** — są łączone w dwóch opaskach, mocujących przewód do goleni. Na przykład pierwszy segment — **X.612.R.Curve.001** (Rysunek 7.3.22b) — ma swój początek (i punkt odniesienia) we wtyczce do tarczy hamulca. Drugi koniec dotyka opaski ograniczeniem **Stretch To**. Aby przewód ugiął się elastycznie przy ruchu amortyzatora, jest powiązany z opaską ograniczeniem **Stretch To**. Wzdłuż krzywej wygięciem długi i cienki cylinder **B.612.R.HLine.001**, który udaje przewód (Rysunek 7.3.22c)

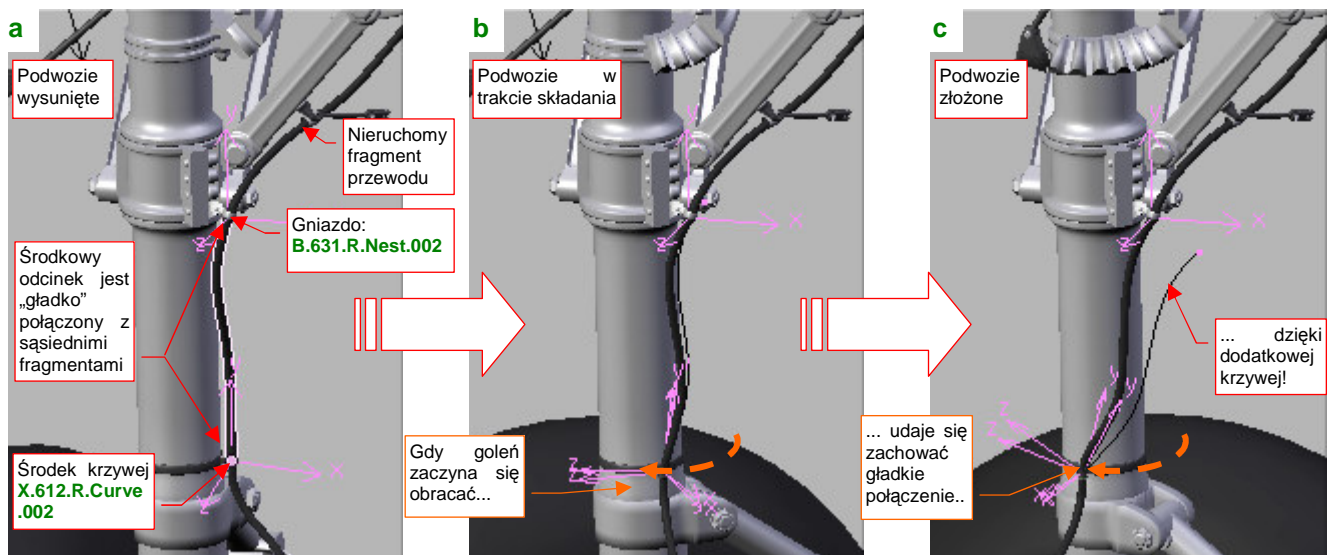
Sprawdźmy, jak „pracuje” przygotowany w ten sposób przewód podczas ruchu amortyzatora (Rysunek 7.3.23):



Rysunek 7.3.23 Ugięcie dolnej części przewodu hamulcowego

Gdy amortyzator jest zupełnie rozciągnięty (bez obciążenia — Rysunek 7.3.23a), ograniczenie *Stretch To* wydłuża wielobok sterujący krzywej **X.612.R.Curve.001**. Odpowiedniemu wydłużeniu ulega także powiązany z nią przewód — przekrój poprzeczny w jego środkowej części staje się wtedy nieznacznie eliptyczny. Tym niemniej nie jest to specjalnie widoczne. Przy ugięciu statycznym amortyzatora (Rysunek 7.3.23b) wszystko wygląda normalnie (bo jest to pozycja „neutralna” modyfikatora *Stretch To*). Przy maksymalnym ugięciu (Rysunek 7.3.23c) na przewodzie pojawiają się nierealistyczne zgrubienie. Zdecydowałem się jednak tak to pozostawić, gdyż w tej pozycji amortyzator będzie się znajdował bardzo rzadko, i tylko przez chwilę. Zresztą w ujęciach z większych odległości ten defekt w ogóle nie będzie widoczny.

Podobnie chciałem zamodelować środkową część przewodu (**X.612.R.Curve.002**). Jej górny kraniec był przytworzony do nieruchomej „opaski”, a dolny — obracał się razem z gołenią (Rysunek 7.3.24):

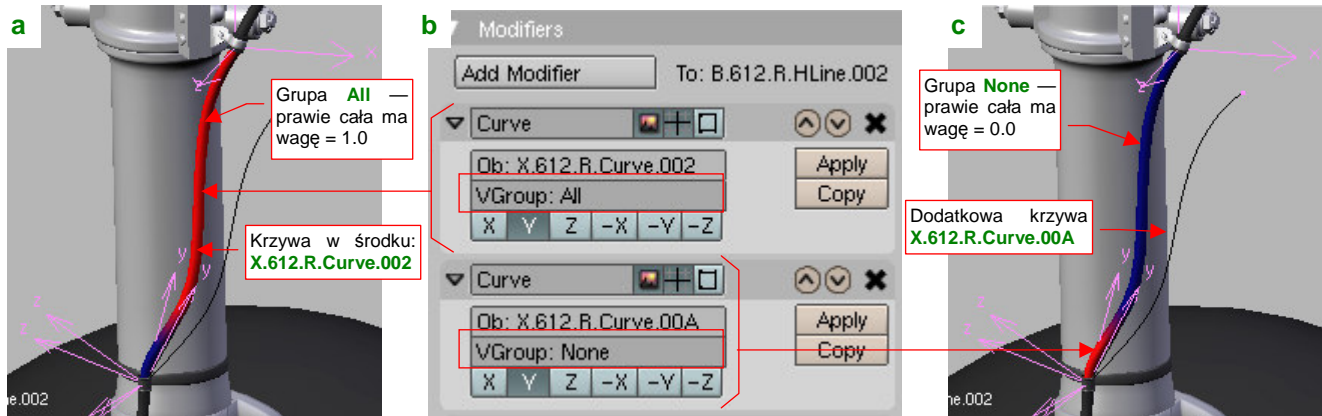


Rysunek 7.3.24 Skręcenie środkowej części przewodu

Jak przystało na elastyczny, ciągły kabel, końcówki naszego przewodu powinny zawsze przechodzić „gładko” w odcinki sąsiednie. Krzywa **X.612.R.Curve.002** ma środek na dolnym krańcu, i jest „przytworzona” (relacją *Parent*) do obracanej części gołeni. Podobnie jak poprzedni fragment, jest „rozciągana” ograniczeniem *Stretch To*, wzdłuż osi Y, do nieruchomego gniazda **B.631.R.Nest.002** (Rysunek 7.3.24a). Okazało się jednak, że, trzeba użyć drugiej krzywej, aby zachować w miarę gładki styk z dolną częścią przewodu (Rysunek 7.3.24b,c)!

W ferworze pracy, dopiero po znalezieniu rozwiązania uświadomiłem sobie, że po złożeniu podwozia jest to miejsce zupełnie zasłonięte pokrywą ☹. Zdecydowałem się jednak opisać tutaj szczegóły tego efektu. To rozwiązanie jest dość skomplikowane i nie jestem z niego specjalnie dumny, ale jest przykładem sterowania modyfikatorami za pomocą wag (*weight*).

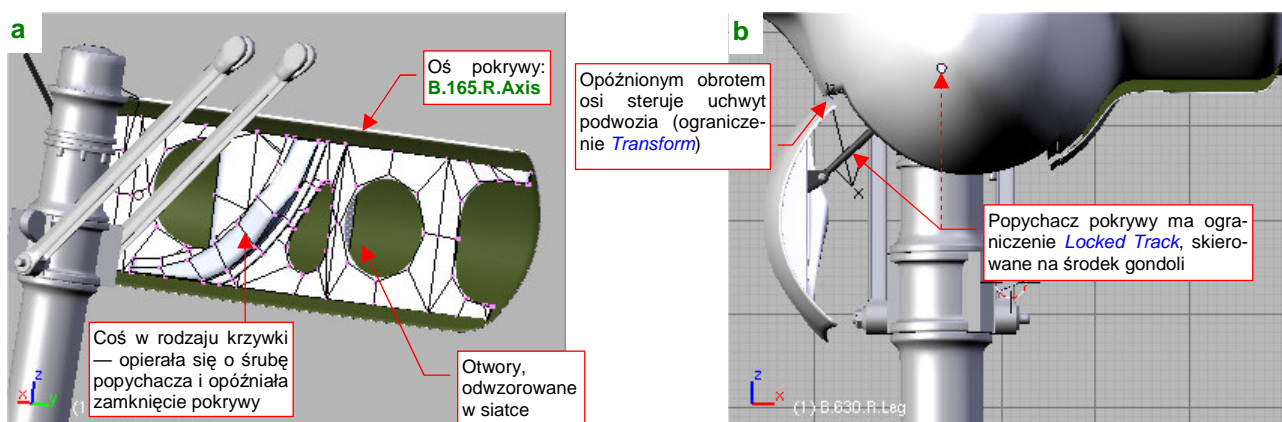
Rysunek 7.3.25 przedstawia „wykrzywiony” przewód na zupełnie obróconej goleni:



Rysunek 7.3.25 Szczegóły deformacji środkowej części przewodu (tryb *Weight Paint*)

W środku przewodu znajduje się podstawowa krzywa: **X.612.R.Curve.002** (Rysunek 7.3.25a). Siatka kabla — **B.612.Hline.002** — ma zdefiniowaną grupę wierzchołków o nazwie **All**. „Pomalowałem” (w trybie *Weight Paint*) prawie całą powierzchnię tej grupy współczynnikiem wpływu (*weight* = 1.0), stąd ma kolor czerwony. (Więcej o trybie *Weight Paint* i współczynnikach wpływu i ich malowaniu znajdziesz na str. 919). Tylko sam dolny koniec pozostawiłem „nieczuły” na wpływ krzywej (*weight* = 0.0 to kolor niebieski). Grupa **All** jest przypisana w panelu modyfikatora *Curve* krzywej **X.612.R.Curve.002** (pole *VGroup*), przez co większość kabla wygina się zgodnie z jej kształtem (Rysunek 7.3.25b). Wygięciem dolnego końca steruje druga krzywa — **X.612.R.Curve.00A**. Ma taki sam kształt, jak pierwsza. Przypisałem jej jednak odwrotne wagi, związane z grupą **None**. W tej grupie waga = 1.0 występuje tylko na samym końcu przewodu. Aby zapewnić właściwy kształt kabla, pomocnicza krzywa **X.612.R.Curve.00A** pokrywa się z **X.612.R.Curve.002** gdy podwozie jest wysunięte (por. Rysunek 7.3.24a). Gdy goleń zaczyna się obracać, krzywa **X.612.R.Curve.00A** zaczyna się odchyłać (ma przypisane odpowiednie ograniczenie *Transform*). Dobrałem jej ruch w ten sposób, by w krańcowym położeniu jej odchylenie wyginało końcówkę kabla we właściwą stronę („gładkiego” połączenia z resztą).

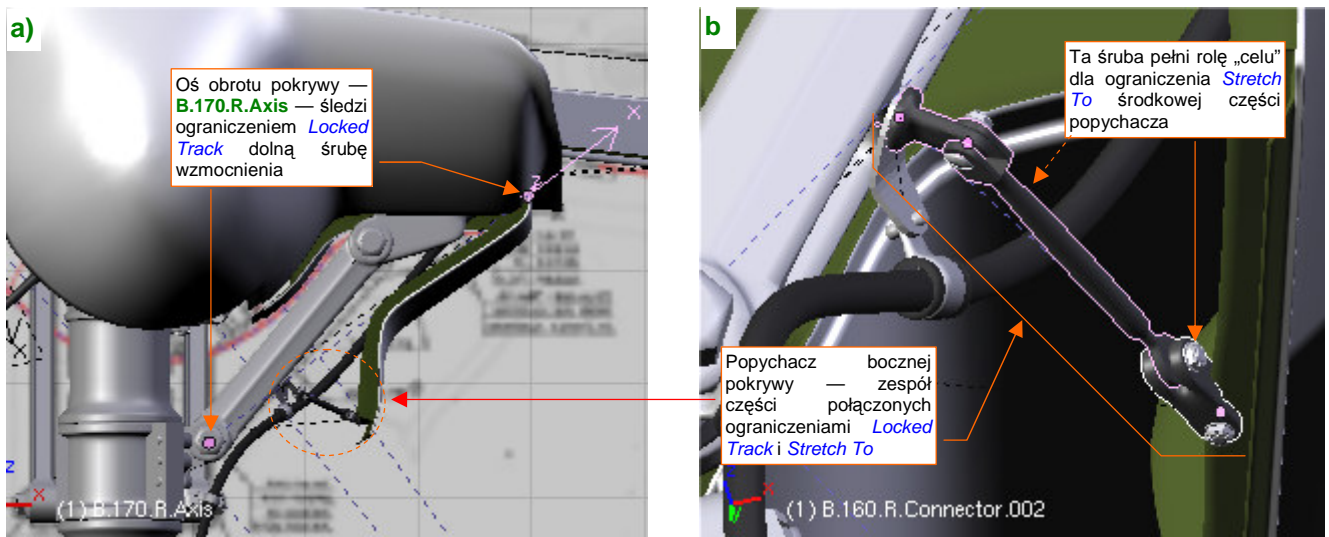
Kolejnym elementem zespołu podwozia jest pokrywa goleni (Rysunek 7.3.26):



Rysunek 7.3.26 Pokrywa goleni wraz z popychaczem

Przy otwartym podwoziu jej otwory są dość widoczne, więc zdecydowałem się je wykonać nie teksturą przejrzystości, a „w siatce” (Rysunek 7.3.26a). Najważniejszą częścią pokrywy jest oś (**B.165.R.Axis**). Przypisałem jej ograniczenie *Transform*, by zamykała się w odpowiedni sposób. Popychacz i jego okucie (Rysunek 7.3.26b) to przerobione siatki z pokryw kółka ogonowego. Popychacz ma przypisane ograniczenie *Locked Track*.

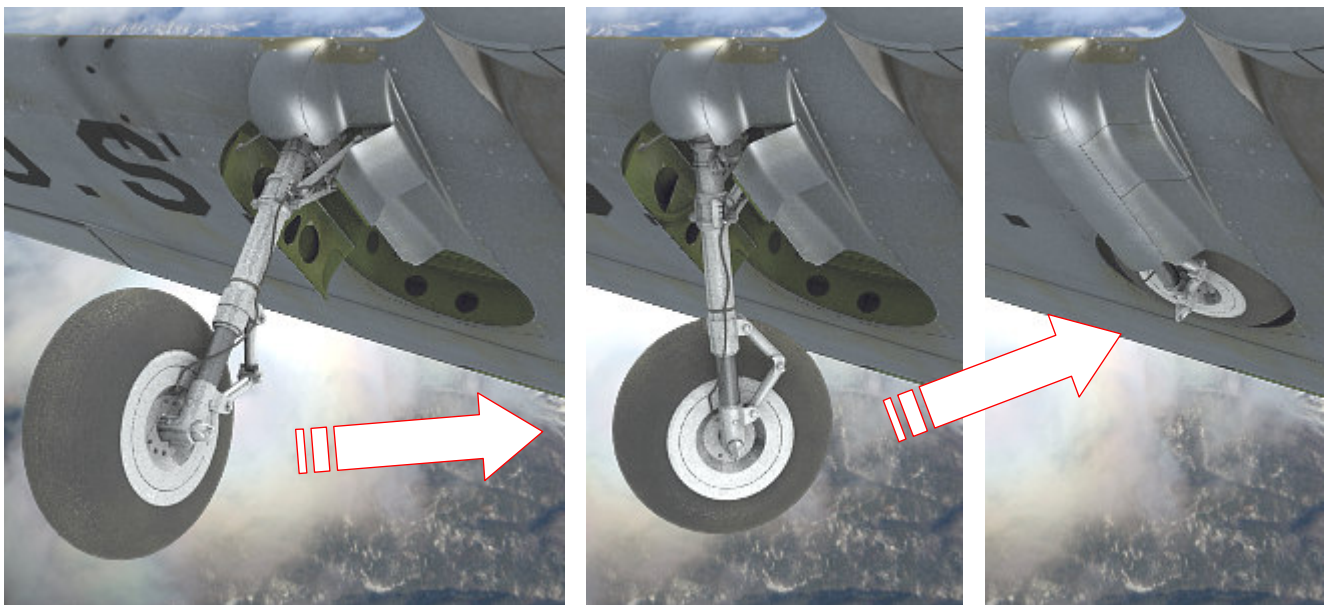
Oś pokrywy bocznej obraca się w trakcie wysuwania goleni dzięki ograniczeniu *Locked Track*, kierującemu oś **X** na jedną ze śrub<sup>1</sup> (Rysunek 7.3.27a):



Rysunek 7.3.27 Pokrywa boczna i jej popychacz

Wraz z osią obraca się połączona z nią „na sztywno” (relacją *Parent*) pokrywa. Pomiędzy pokrywą a golenią podwozia znajdował się mały popychacz (Rysunek 7.3.27b). Powstał ze złożenia trzech śrub i tyłu elementów „podłużnych”. (To znacznie przerobiony fragment popychacza pokryw kółka ogonowego — por. str. 443, Rysunek 7.2.25). Śruby w tym elemencie nie są tylko „upiększeniem”. Pełnią także rolę „celów” dla poszczególnych ograniczeń *Locked Track* i *Stretch To*. W ten sposób unikamy komunikatów w konsoli Blendera o wykryciu zapętlenia w zależnościach pomiędzy obiektami. (Z tego samego powodu dodałem podobne śruby do mechanizmu kółka ogonowego).

Rysunek 7.3.28 prezentuje ostateczny efekt — podwozie główne w kolejnych fazach składania:



Rysunek 7.3.28 Podwozie główne podczas wciągania

<sup>1</sup> Gdyby na przedłużeniu osi **X** nie leżał środek żadnej ze śrub, odłączyłbym ją na chwilę od pokrywy, i odpowiednio obrócić, kierując oś **X** na właściwy cel. Należy to robić, gdy podwozie jest złożone, a pokrywa - domknięta. Cel musi być tak dobrany, by przy wysuniętym podwoziu oś **X** odchyliła się o taki kąt, o jaki obracała się pokrywa. Gdyby w modelu nie byłoby na co wycelować osi — dodałbym odpowiedni obiekt *Empty*. Potem powtórnie przypisałbym pokrywę do osi relacją *Parent*.

### Podsumownie

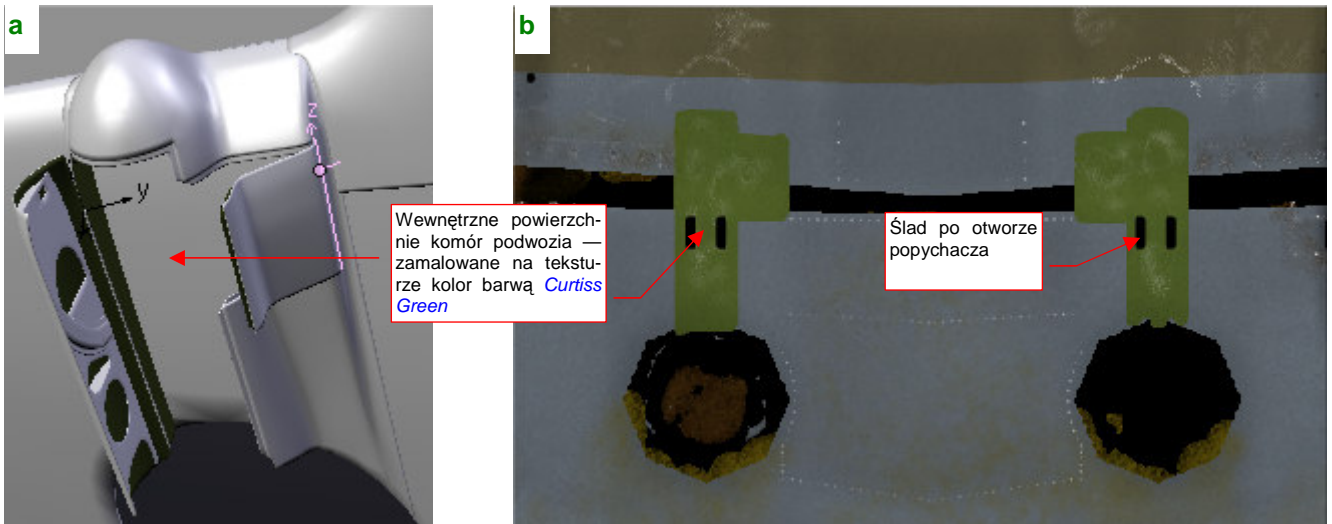
- Dokładne rysunki podwozia samolotu często nie są dostępne. Należy wówczas posłużyć się zdjęciami. Zazwyczaj najlepszej jakości, kolorowe ujęcia pochodzą ze współcześnie odrestaurowanych egzemplarzy. Takie „maszyn z odzysku” powinienś traktować z pełną rezerwą: w wielu szczegółach mogą się różnić od oryginału! (Zazwyczaj dlatego, że nie zachowały się odpowiednie detale, a ich wierne odtworzenie było zbyt drogie);
- Opony koła głównego i kółka ogonowego można pokryć tym samym materiałem (w tym modelu jest to **B.Rubber.Tyre** str. 445). Gdyby koło główne miało bieżnik (a tak jest w późniejszych wersjach P-40) — trzeba byłoby stworzyć nowy materiał, z odpowiednią teksturą nierówności. Można byłoby jednak w nim powtórnie wykorzystać przynajmniej większość tekstur proceduralnych, które opracowaliśmy w poprzedniej sekcji;
- Podwozie główne jest modelowane w ten sam sposób, co kółko ogonowe (str. 446 - 447). Czasami, by uniknąć kłopotliwego łączenia siatek, „łączyłem” dwie powłoki poprzez zwykłe złożenie (str. 449);
- Do uzyskania realistycznego ruchu ramion łącznika można użyć ograniczenia **Limit Distance** (str. 450);
- Do „uruchomienia” podwozia używany zespół dwóch uchwytów — pomocniczego do ugięcia amortyzatora i głównego do chowania i otwierania (str. 451). Takie samo rozwiązanie zastosowaliśmy w poprzedniej sekcji, do wymuszenia ruchu kółka ogonowego;
- Koło podwozia musi być prawidłowo dopasowane do swojej komory (str. 452). Ten proces wymaga zazwyczaj kilku przybliżeń. Czasami można odkryć, że opona jest zbyt szeroka, lub amortyzator miał w pozycji „swobodnej” trochę inną długość;
- Okucia, mocujące podwozie, i inne detale mogą przysporzyć dużo pracy, jeżeli chcesz je wykonać dokładnie. Lepiej uważaj, i naucz się w odpowiednim momencie rezygnować z części drobnych, mało widocznych szczegółów!
- Do odtworzenia przewodu, doprowadzającego płynem hamulcowym do tarczy koła, najlepiej jest skorzystać z krzywej (**Path**) i modyfikatora **Curve** (str. 454);



## 7.4 Podwozie — wykończenie

W poprzednich sekcjach uformowaliśmy i uruchomiliśmy kółko ogonowe i jedno z kół głównych. W tej sekcji uzupełniamy wszystko, czego jeszcze w podwoziu brakuje, by definitywnie zakończyć ten zespół.

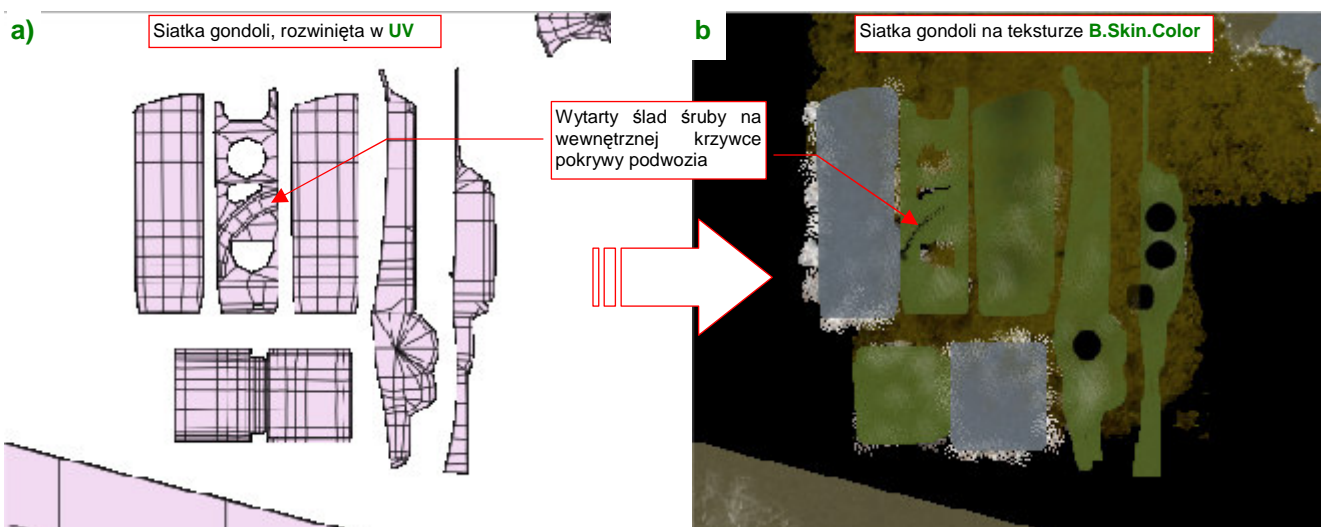
Zacznijmy od wnętrza komory podwozia. W przypadku kółka ogonowego wypełniliśmy ją wręgami już na etapie modelowania (por. str. 259). Uważam, że ten fragment jest niemal na każdym ujęciu niewidoczny, więc nie trzeba go o nic dodatkowego wzbogacać. Co innego z wnętrzem gondoli koła głównego (Rysunek 7.4.1a):



Rysunek 7.4.1 Zamalowane fragmenty płata wewnątrz owiewek goleni podwozia

W P-40 była to wzmocniona powierzchnia płata, pokryta taką samą farbą jak inne powierzchnie wewnętrzne: **Curtiss Green**. Uznałem, że nie trzeba w tym miejscu zagęszczać siatki skrzydła tylko po to, by przypisać odpowiednim ścianom materiał **B.Skin.Inner**. Tak więc ten fragment jest pokryty materiałem **B.Skin.Camuflage**, tak jak cała reszta płata. Po prostu zamalowałem odpowiednie miejsca tekstury barwą (*color.png*) odpowiednim kolorem (Rysunek 7.4.1b).

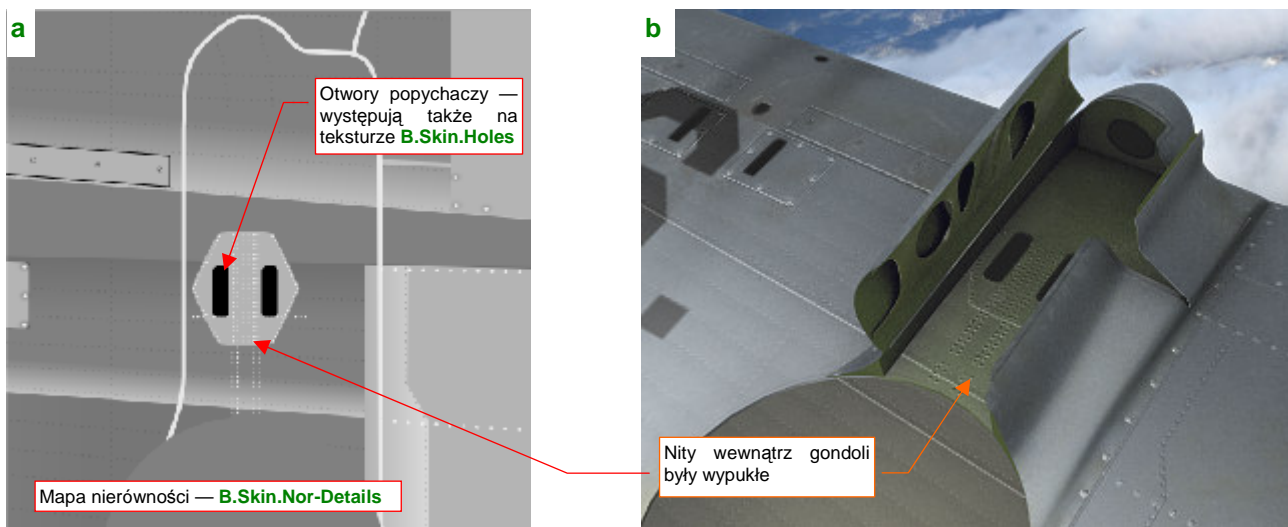
Na tym samym obrazie zamalowałem barwą **Curtiss Green** ścianki boczne gondoli (obydwie wykorzystują tę samą siatkę, stąd na powierzchni tekstury tylko jedno rozwinięcie UV (Rysunek 7.4.2a):



Rysunek 7.4.2 Zamalowane rozwinięcia owiewki goleni podwozia

Co prawda kilka z tych ścian jest i tak pokrytych materiałem **B.Skin.Inner**, który nie wykorzystuje tekstury barw. Zamalowałem je jednak „przy okazji” — a nuż kiedyś się przyda? Najważniejsze było namalowanie zabrudzeń na wewnętrznej powierzchni pokrywy podwozia (podobnie jak skrzydło, jest pokryta **B.Skin.Camuflage**).

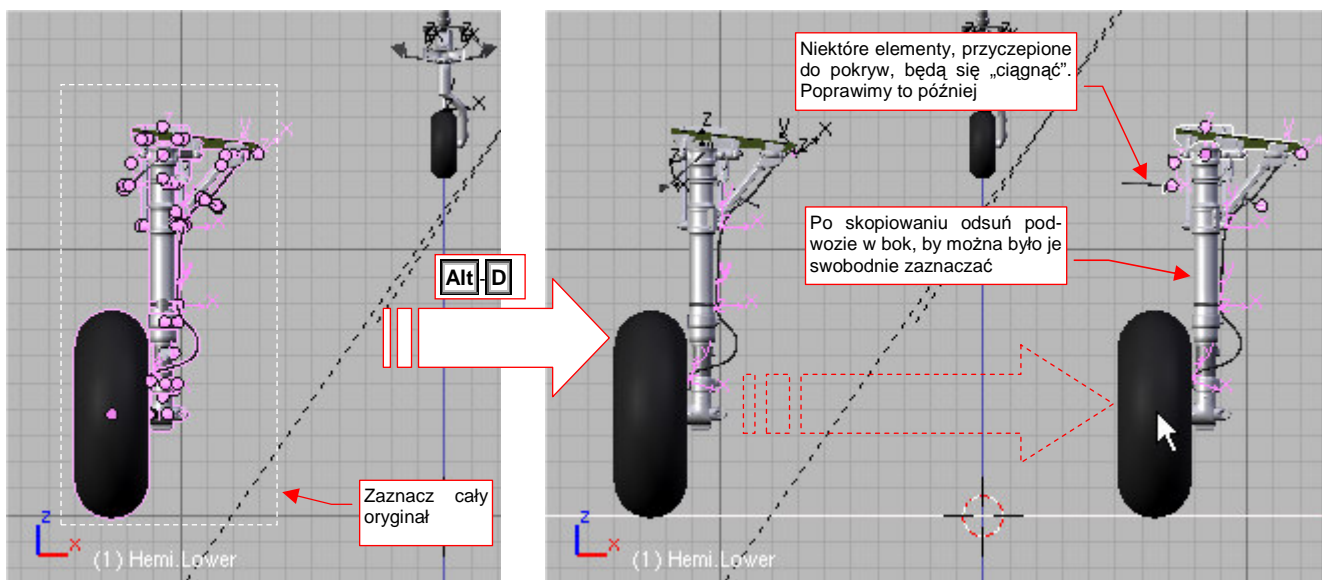
Poza barwą, warto wzbogacić ten fragment poszycia o zarys blach, naniesionych w charakterze wzmocnienia, nity, oraz małe otwory (Rysunek 7.4.3):



Rysunek 7.4.3 Odzworowanie szczegółów powierzchni płata wewnątrz owiewki podwozia

Nity na wszystkich powierzchniach wewnętrznych były wypukłe (bo naniesienie takiego szwu było mniej pracochłonne i tańsze, a sam szew — bardziej wytrzymały). Naniósłem więc zespół białych „kótek” na odpowiednią warstwę rysunku *skin.svg*. Na tym rysunku wektorowym znalazło się także wzmocnienie wokół otworów w skrzydle (Rysunek 7.4.3a), oraz same otwory. Z pliku *skin.svg* powtórnie wygenerowałem ponownie wszystkie pliki rastrowe, by uzyskać nowe obrazy tekstur (zgodnie ze schematem ze str. 418). Rezultat zastosowania tych poprawionych tekstur pokazuje Rysunek 7.4.3b).

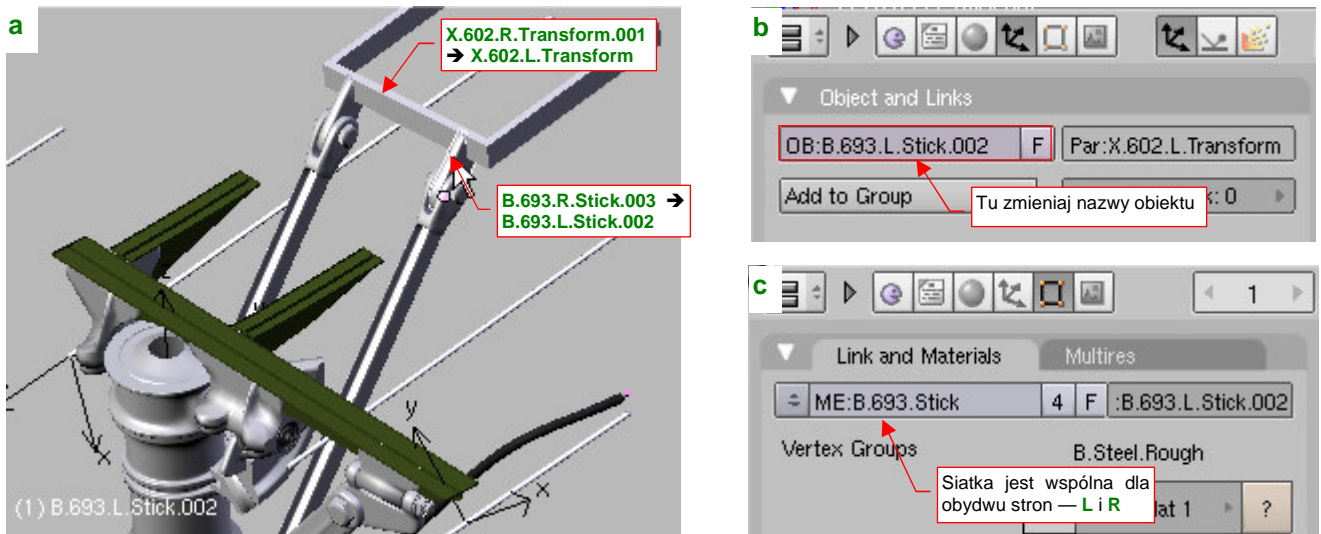
W ten sposób zespół prawego koła głównego jest kompletny i ukończony. Teraz można go skopiować w koło lewe (Rysunek 7.4.4):



Rysunek 7.4.4 Skopiowanie (sklonowanie) podwozia

Jak się przekonałeś w poprzedniej sekcji, kopiowany zespół ma wiele wzajemnie zależnych części. Kilka z nich jest powiązanych z pozostałymi za pomocą ograniczeń *Transform*, *Locked Track*, *Stretch To*. Takich całości lepiej nie powielać poleceniem *Mirror*, bo potem poprawianie rezultatu może zająć dużo czasu. Zamiast tego proponuję zaznaczyć całość i sklonować (**Alt-D**). (W ten sposób obiekty wchodzące w skład lewego koła będą wykorzystywać te same siatki, co koło prawe). Przed zaznaczeniem pamiętaj, aby włączyć także warstwę nr 9, z pomocniczymi obiektami (zawiera krzywe, „wyginające” przewód hydrauliczny).

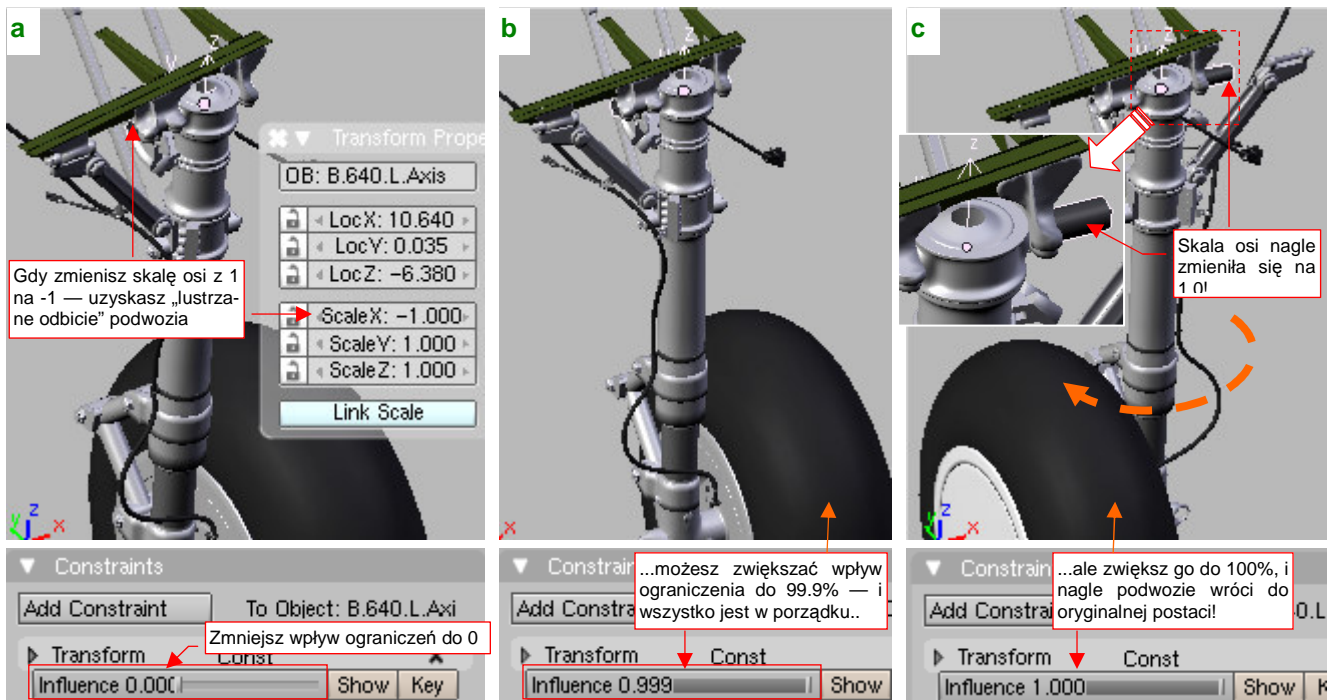
Teraz czeka nas trochę żmudnej pracy „porządkowej”: zmiana nazw wszystkich obiektów, wchodzących w skład skopiowanego zespołu. Blender domyślnie dodał do nich końcówki „001”, „002” (albo niektórym z nich takie końcówki usunął). By mieć porządek w modelu, musimy poprawić wszystkie nazwy, zmieniając w nich symbol strony z „R” na „L”. Aby niczego nie przeoczyć, rób to metodycznie, zaczynając np. od góry (Rysunek 7.4.5):



Rysunek 7.4.5 Porządkowanie nazw

Przesuwaj się sukcesywnie do dołu, aż nie poprawisz nazw całego zespołu. Na koniec sprawdź, czy o czymś nie zapomniałeś, przeglądając ten fragment w oknie *Outliner* (p. str. 58).

Najszybszą metodą na uzyskanie „lustrzanego” odbicia koła prawego jest zmiana skali wzdłuż osi **X** na przeciwną (z 1.0 na -1.0) paru kluczowych elementów. Wystarczy tak przestawić kilka kluczowych elementów, które są „rodzicami” pozostałych. W przypadku naszego podwozia jest to ramka okucia (**B.660.L.Frame**) i oś obrotu goleni (**B.640.L.Axis**). Rysunek 7.4.6a) przedstawia przestawienie skali dla osi obrotu goleni:

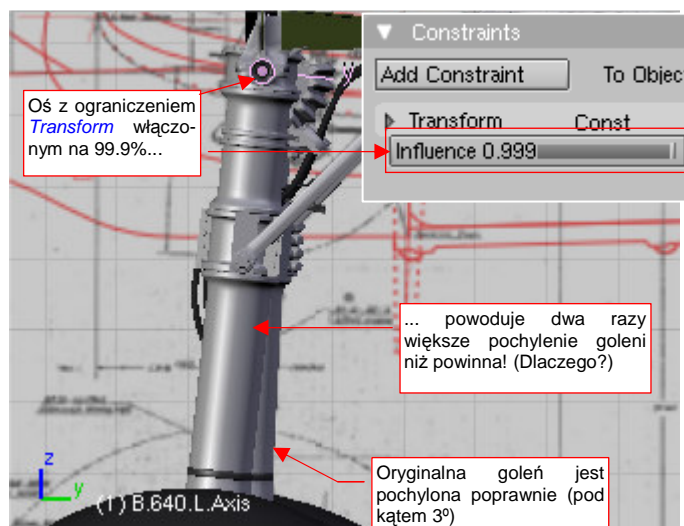


Rysunek 7.4.6 Nieoczekiwane „efekty uboczne” ograniczeń

O ile z ramką okuć nie było problemów, o tyle z osią okucia jest coś nie tak. Dopóki wpływ jej ograniczenia *Transform* jest < 100% (Rysunek 7.4.6b) wygląda prawidłowo, ale po osiągnięciu 100% (Rysunek 7.4.6c) nagle „przeskakuje” na drugą stronę! (Tak, jak gdyby skala **X** osi z powrotem zmieniła się na 1.0)!

Może więc ustawić wpływ ograniczenia *Transform* na 99.9% (Rysunek 7.4.6b)? Różnica transformacji w stosunku do „pełnego” 100% powinna być niezauważalna...

Niestety, to nie koniec niespodzianek! Okazuje się, że po zmianie skali w kierunku **X** na ujemną, transformacja związana z ograniczeniem zaczęła działać dziwnie. Rysunek 7.4.7 pokazuje, jak to wygląda w rzucie z boku. Dla *Influence* = 0.999 oś jest pochylona o dwa razy większy kąt, niż powinna! (Na oko wygląda to na 6° — z tyłu widać prawą goleń, której oś ma wszystkie skale dodane. Ta, pod wpływem takiego samego ograniczenia, jest pochylona pod poprawnym kątem 3°.

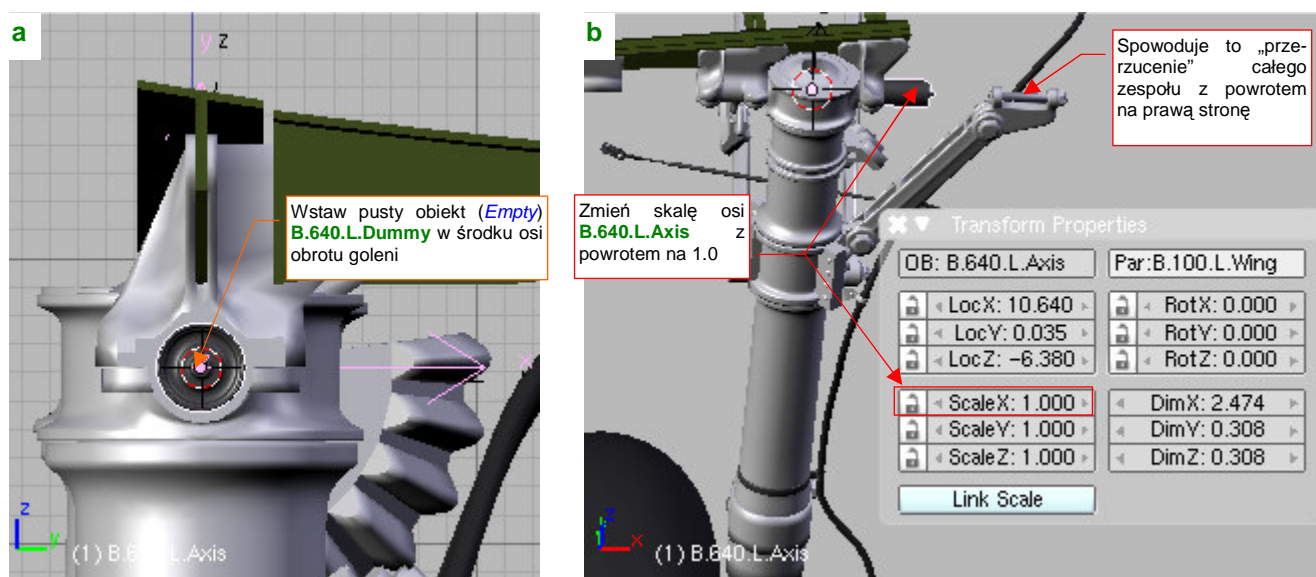


Rysunek 7.4.7 Nieoczekiwane „efekty uboczne” ograniczeń, c.d.

- Wygląda na to, że w Blenderze nie należy stosować ograniczeń dla obiektów, których skala względem jednej z osi jest ujemna. To może być błąd w programie. Jeżeli nawet tak jest, to od dawna nie był poprawiany — pamiętam go od pierwszej wersji, w której pracowałem (2.42, z połowy 2006r).

Jak sobie poradzić z tym problemem? Wystarczy, by ujemną skalę zastosować do jakiegoś dodatkowego obiektu, który „przymocujemy” (relacją *Parent*) do osi. Ten pomocniczy obiekt przejmie jednocześnie rolę „rodzica” wszystkich dalszych elementów, oryginalnie przypisanych do osi obrotu gołeni.

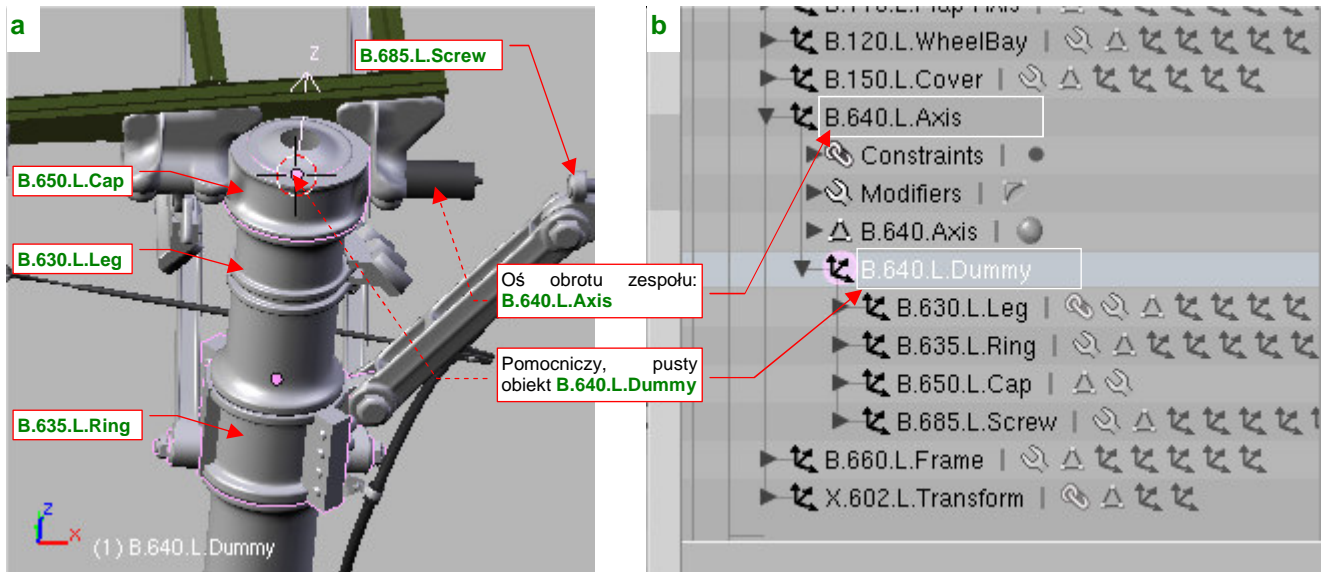
Wyłącz na razie wpływ ograniczenia *Transform* (*Influence* = 0.0). To ustawi cały zespół gołeni pionowo i ułatwi pracę. Ustaw kursor w środku (*origin*) osi **B.640.L.Axis** (najłatwiej zaznaczyć oś i wywołać **Shift-S**, *Cursor→Selection*). Wstaw tam pomocniczy, pusty (*Empty*) obiekt (Rysunek 7.4.8a). Nadaj mu nazwę **B.640.L.Dummy**. Dla pewności, zorientuj jego osie tak, pokrywały się z osiami obiektu **B.640.L.Axis**.



Rysunek 7.4.8 Wstawienie pomocniczego obiektu i wyłączenie ujemnej skali osi

Teraz można zmienić skalę **B.640.L.Axis** w kierunku osi **X** z powrotem na 1.0 (Rysunek 7.4.8b). Spowoduje „odbicie” kształtu całej ruchomej części podwozia z „lewej” na „prawą”. Za chwilę to poprawimy, ale na razie nie będzie pasować do nieruchomych okuć ramki **B.640.L.Frame**.

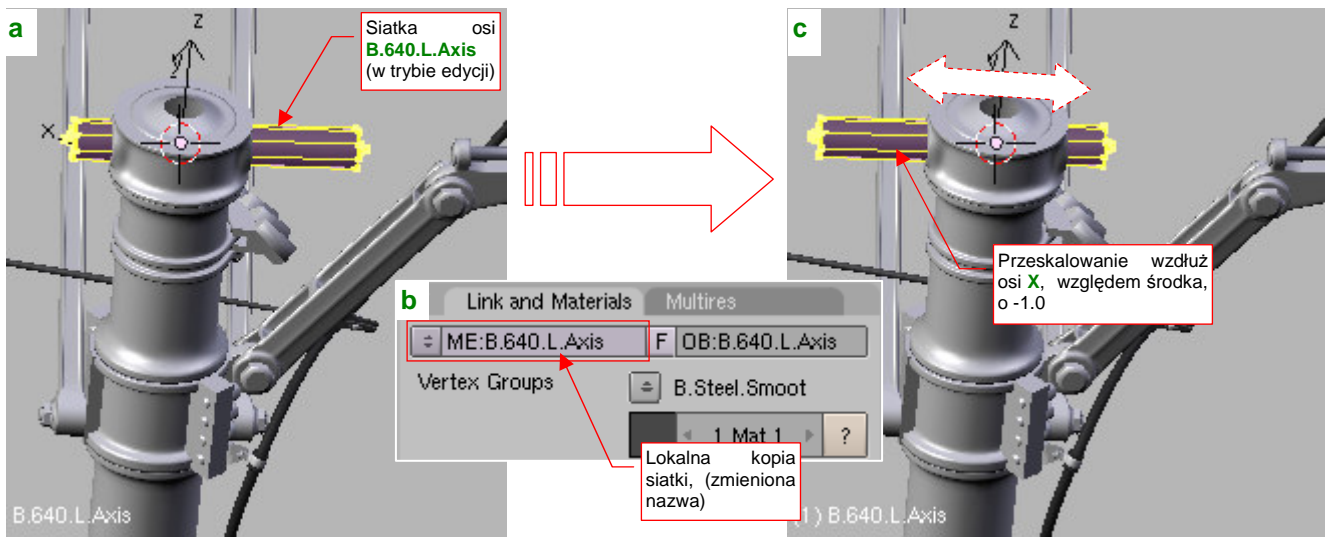
Teraz przypisz (**Ctrl-P**) wszystkim czterech „bezpośrednich potomków” osi **B.640.L.Axis** do pomocniczego obiektu **B.640.L.Dummy** (Rysunek 7.4.9a):



Rysunek 7.4.9 Zmiana hierarchii lewej goleni

Obiekt **B.640.L.Dummy** pozostanie jedynym „bezpośrednim potomkiem” **B.640.L.Axis**. W rezultacie powinieneś uzyskać taką hierarchię, jaką pokazuje Rysunek 7.4.9b).

Możemy się już zająć kształtem osi zespołu — obiektu **B.640.L.Axis**. Przejdź do trybu edycji jej siatki (Rysunek 7.4.10a):

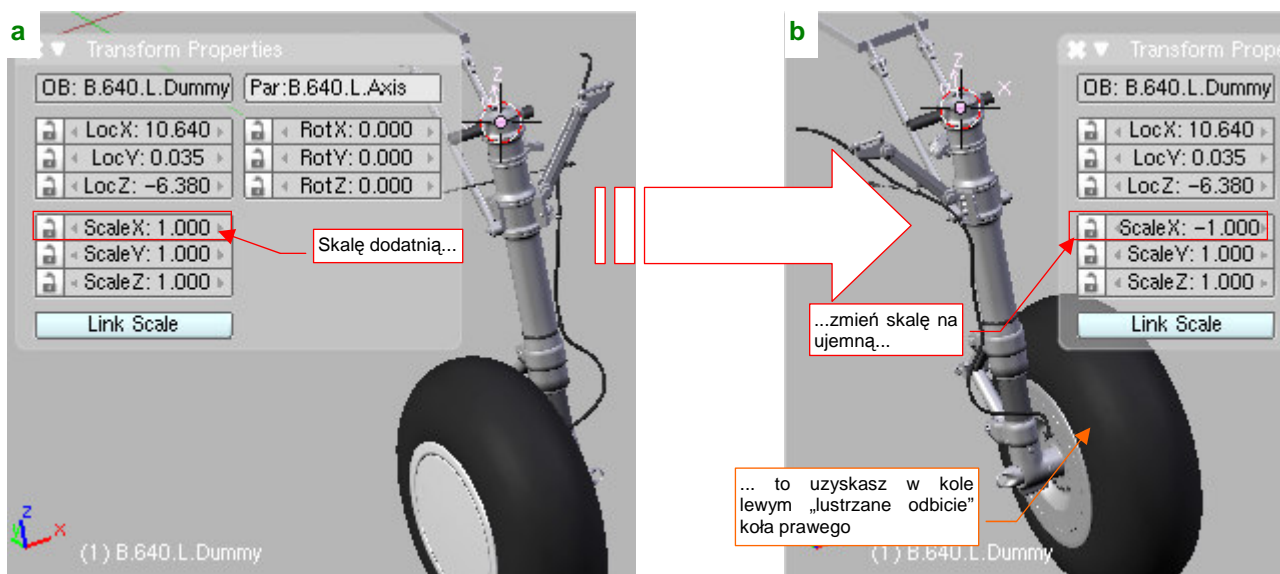


Rysunek 7.4.10 Modyfikacja siatki osi (odbicie lustrzane)

Skoro nie możemy zmienić skali tego obiektu na ujemną, zmienimy jego kształt, przeskalowując w odpowiednim kierunku wierzchołki. Nim to zrobisz, w panelu *Link and Materials* (zestaw *Editing*) zmień nazwę jego siatki na **B.640.R.Axis** (nie zapominaj, że współdzielił ją z prawą golenią!). Potem skopiuj tę siatkę w nową, i nadaj jej nazwę **B.640.L.Axis** (Rysunek 7.4.10b). Teraz możesz już ją zmieniać.

Umieść kursor w środku **B.640.L.Axis** (nie chodzi tu o środek geometryczny, a punkt odniesienia — *origin*). Zaznacz wszystkie wierzchołki jego siatki, i przeskaluj je wzdłuż osi **X**, względem kursora 3D, o wartość = -1.0. To nada z powrotem osi obrotu zespołu odpowiedni kształt, pasujący do okuć goleni (Rysunek 7.4.10c).

Wystarczy już tylko zmienić skalę X pomocniczego obiektu **B.640.L.Dummy**, by uzyskać poprawny zespół lewego koła (Rysunek 7.4.11):

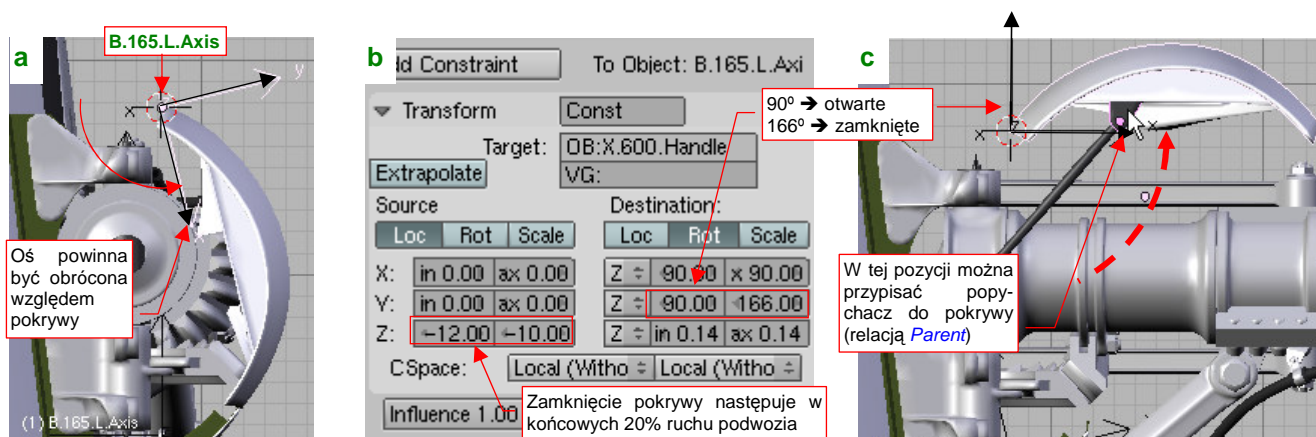


Rysunek 7.4.11 Zmiana znaku skali pomocniczego obiektu

Dzięki „rozdzieleniu ról”: inny obiekt ma przypisane ograniczenie *Transform*, a inny — ujemną skalę, wszystko w mechanizmie lewego koła zaczęło działać poprawnie. No, właściwie z jednym wyjątkiem: pomocniczej krzywej **X.612.L.Curve.A**, odpowiedzialnej za dodatkową deformację środkowego segmentu przewodu (por. str. 456). Jej kształt także uległ nieoczekiwanej zmianie skali, gdy udział (*Influence*) transformacji = 100%. Na szczęście w tym przypadku wystarczyło zmniejszyć ten parametr do 99.9%.

Teraz pozostało „uruchomić” osie pokryw lewego podwozia. (Przewidując problemy z ujemną i ograniczeniami, każda z tych osi jest obiektem „pozytywnym”, gdzie wszystkie skale są dodatnie).

Zacznij od dopasowania „pozycji spoczynkowych”, jeszcze przed przypisaniem ograniczeń. Najlepiej to zrobić, gdy podwozie jest złożone (tak, przekonaj się sam: obydwie golenie już się składają!). Osie będą musiały zostać nieco obrócone względem pokryw. Posługując się współrzędnymi osi prawej pokrywy jako wzorcem, popraw obrót pokrywy lewej (**B.165.L.Axis**, Rysunek 7.4.12a):

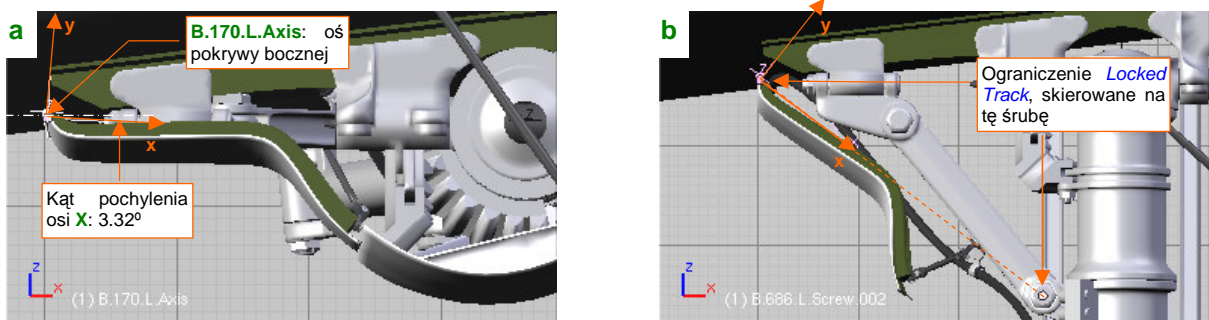


Rysunek 7.4.12 Regulacja osi obrotu pokrywy podwozia (uwaga: skrzydło na ilustracji jest ustawione pionowo)

Potem przypisz do **B.165.L.Axis** ograniczenie *Transform*, uzależniające jej obrót wokół osi **Y** od głównego uchwytu **X.600.Handle** (Rysunek 7.4.12b). Zwróć uwagę, że kąty obrotu tego ograniczenia są nieco inne niż w prawej, „wzorcowej” osi (**B.165.R.Axis**). Po prostu obydwie pokrywy obracają się w przeciwnie strony.

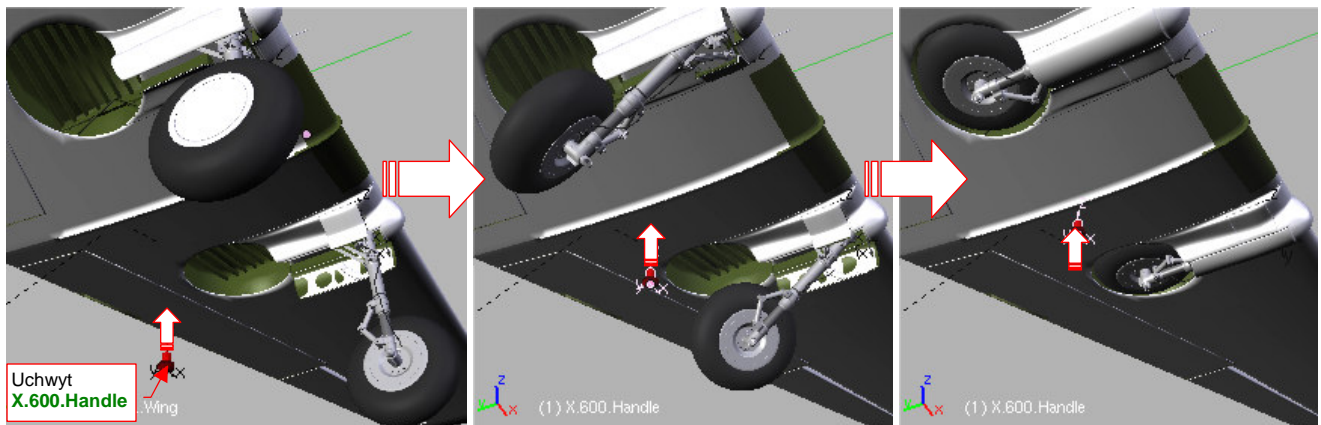
Teraz możesz już sprawdzić, jak pokrywa się otwiera. Wsuń podwozie (Rysunek 7.4.12c)! Przy okazji, gdy pokrywa goleni znajdzie się w położeniu „otwarte”, przypisz do niej (relacją *Parent*) popychacz. (Ta kopia do tej pory nie miała „rodzica”, więc się w ogóle nie poruszała).

Analogicznie jak pokrywę goleni, „uruchom” pokrywę boczną. Zacznij od dopasowania „spoczynkowego” kąta osi obrotu (gdy podwozie jest zamknięte) (Rysunek 7.4.13a), a potem przypisz jej odp. ograniczenie *Locked Track* (Rysunek 7.4.13b):



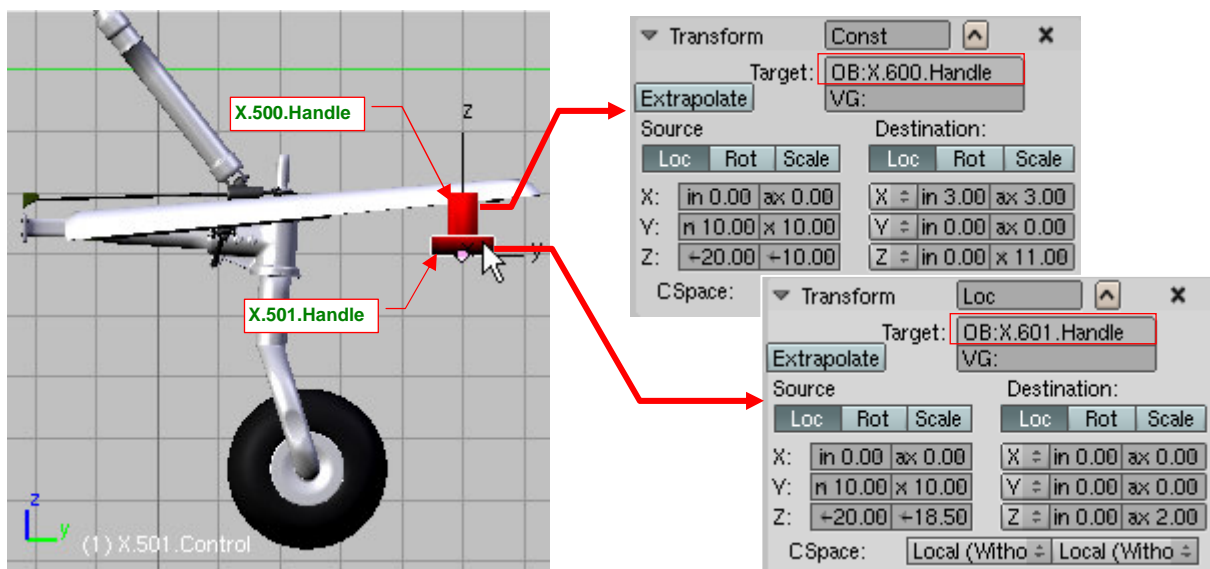
Rysunek 7.4.13 Dopasowanie zakresu kątów pokrywy bocznej

Gdy przypiszesz już jej wszystkie ograniczenia i popychacz, można sprawdzić działanie pełnego zespołu podwozia głównego (Rysunek 7.4.14):



Rysunek 7.4.14 Test składania całego podwozia głównego

Składa się całkiem realistycznie, nieprawdą? Skoro tak, to po co nam dwie pary uchwytów — jeden do kół głównych, a drugi do kółka ogonowego? Przypisz im odpowiednie ograniczenia *Transform*, wiążące ich ruch z ruchem uchwytów *X.600.Handle* i *X.601.Handle* (Rysunek 7.4.11):

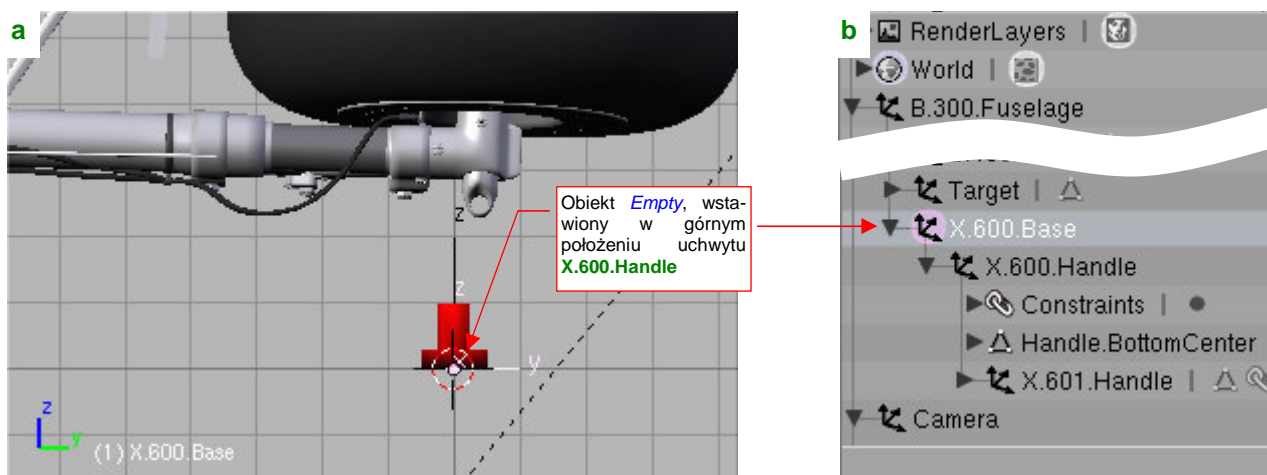


Rysunek 7.4.15 „Sprzężenie” mechanizmu kółka ogonowego z uchwytami podwozia głównego

W ten sposób chowaniem i wysuwaniem podwozia naszego modelu steruje jeden uchwyt — *X.600.Handle*, a ugięcie wszystkich amortyzatorów można ustawić za pomocą uchwytu pomocniczego — *X.601.Handle*.

Nim ukończymy ten model, takich uchwytów jak podwozia będzie potrzeba więcej: do sterów, lotek, klap... Przydałoby się do nich jakaś „panel sterowania”, aby nie trzeba było ich szukać po przestrzeni modelu.

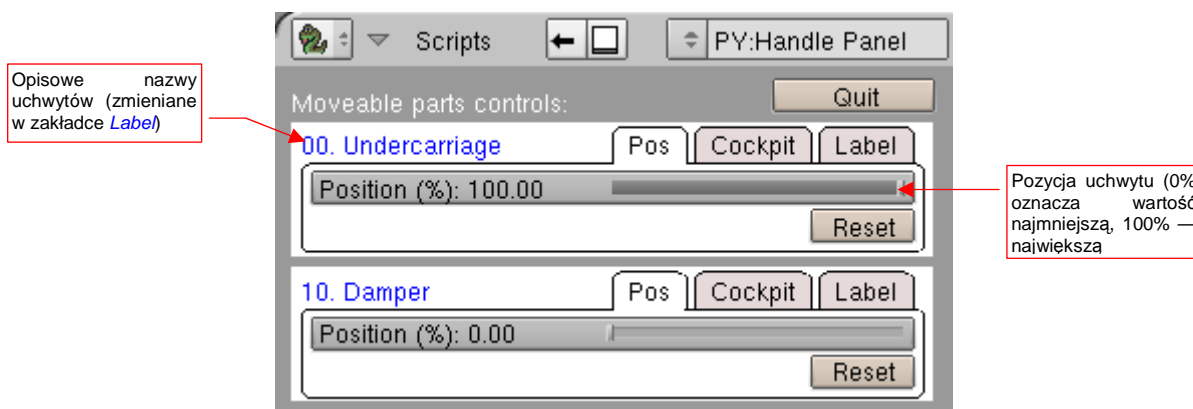
Przygotowując sobie warsztat pracy w Blenderze, parę lat temu stworzyłem właśnie taką uniwersalną panel sterowania modelem. Jest to skrypt *HandlePanel.py* (dołączony do materiałów, towarzyszących tej książce, w folderze *source/python*). Szczegółowy opis, jak się nim posługiwać, znajdziesz na str. 845. Skrypt jest uniwersalny, bo samoczynnie wykrywa w pliku modelu wszystkie uchwyty. Muszą to być po prostu obiekty o nazwie zakończonej przyrostkiem *\*.Handle*, z przypisanym ograniczeniem *Limit Location* (por. str. 846). Muszą być także przypisane (relacją *Parent*) do jakiegoś obiektu nadrzędnego („rodzica”). Środek takiego obiektu jest traktowany przez skrypt jako „położenie domyślne” uchwytu. Załóżmy, że domyślnie podwozie ma być schowane. Stwórz w górnym położeniu uchwytu *X.600.Handle* nowy, pusty (*Empty*) obiekt (Rysunek 7.4.16a), i nadaj mu nazwę *X.600.Base*. Przypisz *X.600.Handle* do *X.600.Base*, a *X.600.Base* — do kadłuba (*B.300.Fuselage*). Hierarchia tego fragmentu powinna wyglądać tak, jak to pokazuje Rysunek 7.4.16b:



Rysunek 7.4.16 Przywiązanie do uchwytu „punktu odniesienia”

Wygląda to w tej chwili tak, że „rodzicem” pomocniczego obiektu, sterującego ugięciem amortyzatora (*B.601.Handle*) jest główny uchwyt podwozia (*B.600.Handle*). W efekcie domyślnym położeniem amortyzatora jest pełne rozciągnięcie (w tej pozycji były rzeczywiście chowane).

Położenie obiektów *X.500.Handle* i *X.501.Handle*, sterujących położeniem kółka ogonowego, uzależniliśmy od uchwytów podwozia głównego. Aby te obiekty nie zostały zidentyfikowane przez skrypt jako uchwyty (ich położenie jest związane z *X.60\*.Handle*), musisz zmienić im nazwę — np. na *X.500.Control* i *X.501.Control*. Na razie panel sterowania samolotem zawiera tylko dwa panele (Rysunek 7.4.17):



Rysunek 7.4.17 Panel sterowania modelem (na razie tylko dwa uchwyty podwozia)

Dzięki panelowi sterowania nie musisz się zastanawiać, do czego służy konkretny uchwyt. (Parę miesięcy po zakończeniu modelu możesz już mieć taki dylemat). W oknie *Handle Panel* masz wszystko czytelnie popisywane. Z czasem okno tego skryptu zapełni się panelami różnorodnych uchwytów.



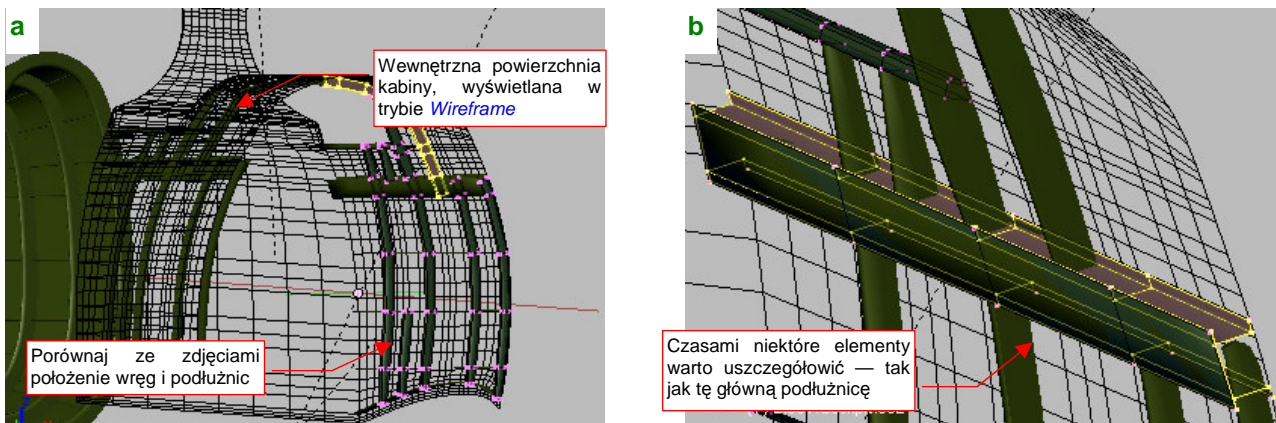
### Podsumownie

- Aby nie komplikować niepotrzebnie siatki, część powierzchni wewnętrznych można namalować wprost na teksturze kamuflażu (str. 459 - 460);
- Zespół drugiego koła (prawego, lewego — w zależności od tego, które wykonałeś jako pierwsze) najlepiej jest stworzyć poprzez sklonowanie (str. 460). Potem rezultat tej operacji wymaga jeszcze trochę pracy, m.in. poprawienia nazw skopiowanych obiektów (str. 461);
- „Lustrzane odbicie” zespołu podwozia najprościej uzyskać, zmieniając skalę na ujemną dla kilku wybranych obiektów (str. 461). (Chodzi zazwyczaj o dwa — trzy elementy, znajdujące się „na szczycie” hierarchii całego zespołu);
- Nie należy zmieniać skali na ujemną dla obiektu, który ma przypisane jakieś ograniczenia (str. 461). W razie czego należy zawsze rozdzielać te role tak, by ujemną skalę przypisać do jakiegoś bezpośredniego potomka elementu z ograniczeniami. W ostateczności można zawsze wstawić w tym celu jakiś dodatkowy obiekt pomocniczy (str. 462 - 464);
- Pokrywy podwozia powstają zazwyczaj wcześniej niż koła i golenie — już podczas modelowania powierzchni samolotu (por. str. 230 i dalsze). Najlepiej dopasować je do reszty zespołu podwozia w pozycji „złożonej” (str. 464 - 465);
- Całe podwozia — główne i ogonowe — można „sprząc” ze sobą tak, by było sterowane pojedynczym zespołem uchwytów (str. 465);
- Na koniec warto uporządkować nazwy wszystkich pomocniczych obiektów. Tylko nazwy „prawdziwych” uchwytów mogą mieć przyrostek „**\*.Handle**”. Dobrze jest także przypisać je (relacją *Parent*) do elementów, których środki wyznaczają „położenie neutralne” (str. 466). To wszystko pozwoli później na prawidłową identyfikację zakresu ruchu całego zespołu przez specjalną „paneli sterowania” (*Handle Panel* — str. 466);

## 7.5 Wnętrze kabiny

Budując wnętrze kabiny, trzeba polegać głównie na zdjęciach - jakiegokolwiek rysunki na planach modelarskich są niemal zawsze za mało dokładne. Podczas budowy modelu stosowaliśmy zasadę „jeżeli czegoś nie odwzorowujesz, to przynajmniej przygotuj tak, aby łatwo można było zrobić to później”. Dzięki niej podczas formowania kadłuba przygotowaliśmy wewnętrzną powłokę kokpitu, wraz z wręgami i podłużnicami (por. str. 222).

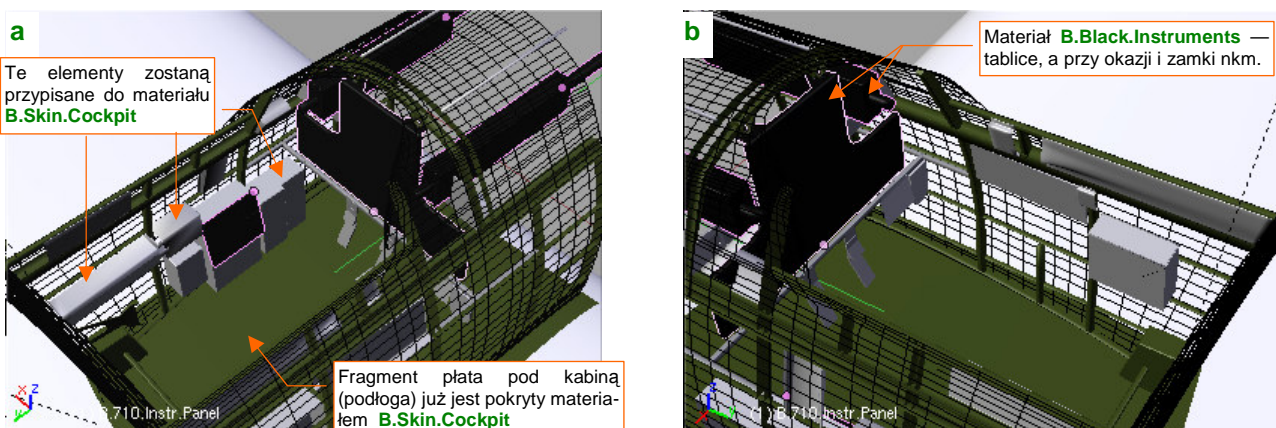
Pracę nad kabiną najlepiej jest zacząć od przełączenia wewnętrznej powłoki z domyślnego trybu wyświetlania *Shaded* w tryb *Wireframe* (zestaw *Object*, panel *Draw*). Ułatwia to znacznie dalsze prace, bo możesz teraz widzieć na wskroś poprzez powierzchnię kadłuba (Rysunek 7.5.1):



Rysunek 7.5.1 Przygotowanie szkieletu kadłuba i poszycia kabiny

Kolejnym krokiem jest powtórne zweryfikowanie kształtu i rozłożenia elementów szkieletu wewnątrz kabiny (Rysunek 7.5.1a). Porównaj je starannie z dostępnymi zdjęciami. To bardzo ważne, bo potem krzyżujące się wręgi i podłużnice będą służyły Ci za coś w rodzaju „siatki odniesienia”. Analizując fotografie, będziesz mógł na ich podstawie szacować wielkość pozostałych elementów wyposażenia. Przy okazji możesz zdecydować się na jakieś uszczegółowienia ich kształtu — na przykład zdecydowałem się zmienić przekrój głównej podłużnicy z „C” na „H”, bo taki widać na zdjęciach (Rysunek 7.5.1b).

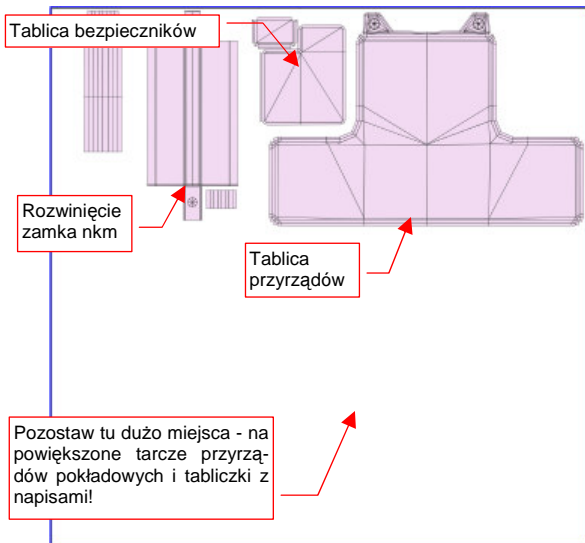
Zdjęcia kabiny przytłaczają liczbą szczegółów. Nie przejmuj się tym tak bardzo! Przyjrzyj się fotografiom i wyodrębni na nich elementy podstawowe (te, do których są przymocowane pozostałe) (Rysunek 7.5.2):



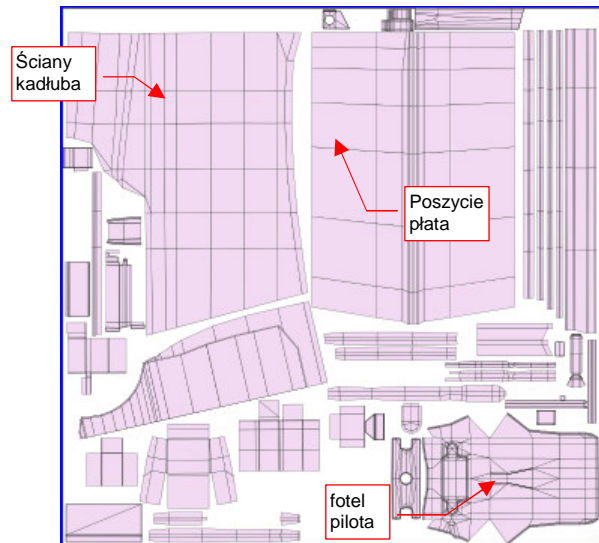
Rysunek 7.5.2 Podstawowe „bryły” wyposażenia kabiny

Zazwyczaj zidentyfikujesz i stworzysz tablicę przyrządów (wraz ze stelażem), oraz parę „pudełek” i „płytek”, ciągnących się wzdłuż lewej (Rysunek 7.5.2a) i prawej (Rysunek 7.5.2b) strony kabiny. Krawędzie „pudełek” warto fazować (poleceniem *Bevel* — por. str. 922). Przygotuj także dwa nowe materiały (możesz je skopiować z *B.Inner.Details*). Pierwszym: *B.Skin.Cockpit*, pokryj „wszystko, co nie jest czarne”. Drugim — *B.Black.Instruments* — pokryjesz „wszystko, co czarne” (tablicę przyrządów, pokrętła, itp.).

Materiał **B.Black.Instruments** będzie miał przypisane oddzielne tekstury, na których odwzorujemy wszystkie drobne szczegóły: tabliczki z napisami, tarcze przyrządów pokładowych. Trochę „na doczepkę” zaliczyłem do nich także wystające w kabinie zamki „półcałówek” Browinga. (Na zdjęciach także są czarne i mają taki sam połysk, co tablica przyrządów). Przygotuj rozwinięcia siatek należących do tego materiału. Aby uzgodnić ich położenie w UV, przypisz wszystkie do pomocniczego obrazu o nazwie **Test-Instruments** (Rysunek 7.5.3):



Rysunek 7.5.3 Rozłożenie w UV siatek przypisanych do materiału **B.Black.Instruments** (obraz **Test-Instruments**).

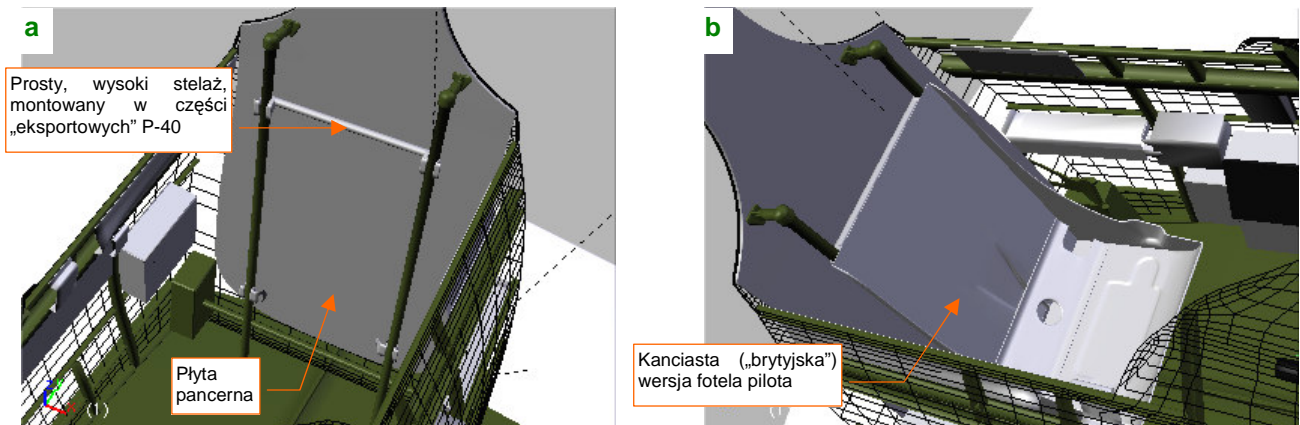


Rysunek 7.5.4 Rozłożenie w UV siatek przypisanych do materiału **B.Skin.Cockpit** (obraz **Test-Cockpit**).

Rozwinięcia ułóż tak, by zostawić na teksturze dużo wolnego miejsca — umieścimy tam powiększone obrazy tarcz instrumentów pokładowych, i różnych innych „opisanych” elementów. (Na przykład: bloku sterowania przepustnicą, mieszanką i skokiem śmigła, pokręteł trymerów, itp.).

Należy oczekiwać, że ciekawscy obserwatorzy będą częściej zaglądać do kabiny niż do luków podwozia. Warto więc zastosować w materiale **B.Skin.Cockpit** odpowiednią mapę nierówności, odwzorowującą łączenia blach i nity, a także mapę barw, odwzorowującą różnorodne przetarcia, zadrapania i zabrudzenia. Aby to było możliwe, rozwiń w **UV** odpowiednie powierzchnie (Rysunek 7.5.4). By nie było ich zbyt dużo, możesz pominąć drobniejsze elementy, przypisując je do materiału **B.Inner.Details**. Dla uproszczenia rozwinąłem tylko symetryczną półkę kadłuba (zabrudzenia będą lustrzanym odbiciem, ale to nie powinno razić).

Zdecydowałem się odwzorować już na tym etapie fotel pilota (Rysunek 7.5.5b). (Jest najbardziej „wyeksponowany”, i na pewno będzie miejscami zabrudzony, więc trzeba go także wkomponować w rozwinięcie **UV**).

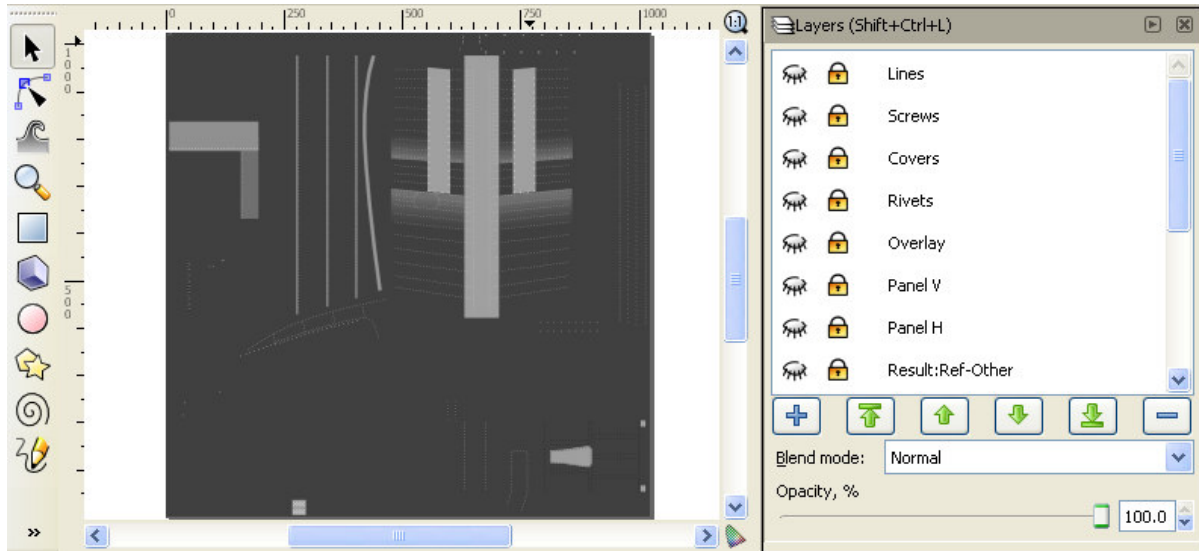


Rysunek 7.5.5 Stelaż i fotel pilota

W P-40 stosowano dwa rodzaje foteli pilota i stelaży, w zależności od odbiorcy: „brytyjski” i „amerykański”. Samoloty AVG miały, wg Hume Batesa ([7]), kanciasty model „brytyjski”, na prostym stelażu (Rysunek 7.5.5a).

Tekstury dla materiału **B.Skin.Cockpit** uzyskamy w ten sam sposób, jak tekstury powierzchni zewnętrznych samolotu (por. str. 371 - 384). Przygotujemy je w oddzielnych plikach, w podkatalogu o nazwie *cockpit*. (Uważam, że oddzielny folder jest wygodniejszym do rozróżniania plików z obrazami tekstur, niż dodawania jakichś przedrostków do ich nazw).

Najpierw wyeksportuj z Blendera rozwinięcia siatek (tych, które pokazywał Rysunek 7.5.4) w formacie SVG, i złóż je w Inkscape (por. str. 949), w pliku *cockpit\cockpit.svg*. Obraz *cockpit\cockpit.svg* będzie mniejszy od obrazu powierzchni zewnętrznych — nadaj mu rozmiar, powiedzmy, 1024x1024px. Potem przygotuj podobne warstwy, jak w pliku *skin.svg* (por. str. 367). (Niektóre możesz pominąć, bo ten rysunek nie będzie miał takich elementów, jak cienie pod nitami lub zagłębienia pod śrubami). Narysuj na nich obraz nierówności wnętrza kabiny (Rysunek 7.5.6):

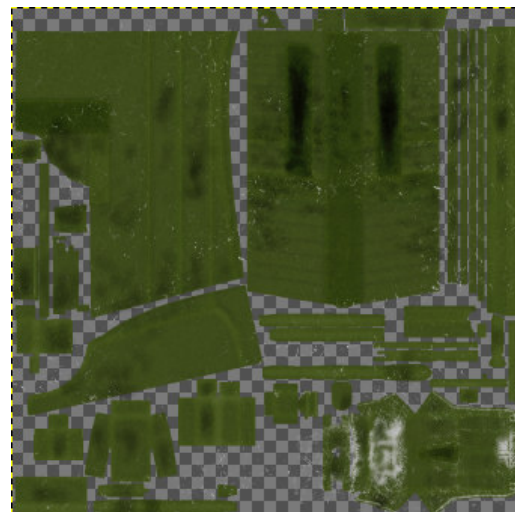


Rysunek 7.5.6 Przygotowywanie w Inkscape tekstury nierówności (plik *cockpit\cockpit.svg*)

Sprawdź w Blenderze, jak ta mapa nierówności „leży” na siatkach, i nanieś odpowiednie poprawki. Gdy będzie gotowa — połącz w Inkscape poszczególne warstwy w grupy (por. str. 371). Skomponuj z klonów tych grup na oddzielnej warstwie obraz map nierówności (**Result:Nor**). Używając odpowiednich filtrów (por. str. 737) przygotuj z klonów tych samych warstw mapę odbić (*ref.png* — por. str. 376-380). Podobnie jak w pliku *skin.svg*, będzie złożenie trzech warstw: **Result:Ref-Blur**, **Result:Ref-Grunge**, **Result:Ref-Other** (Rysunek 7.5.7):



Rysunek 7.5.7 Tekstura odbić (*cockpitref.png*)



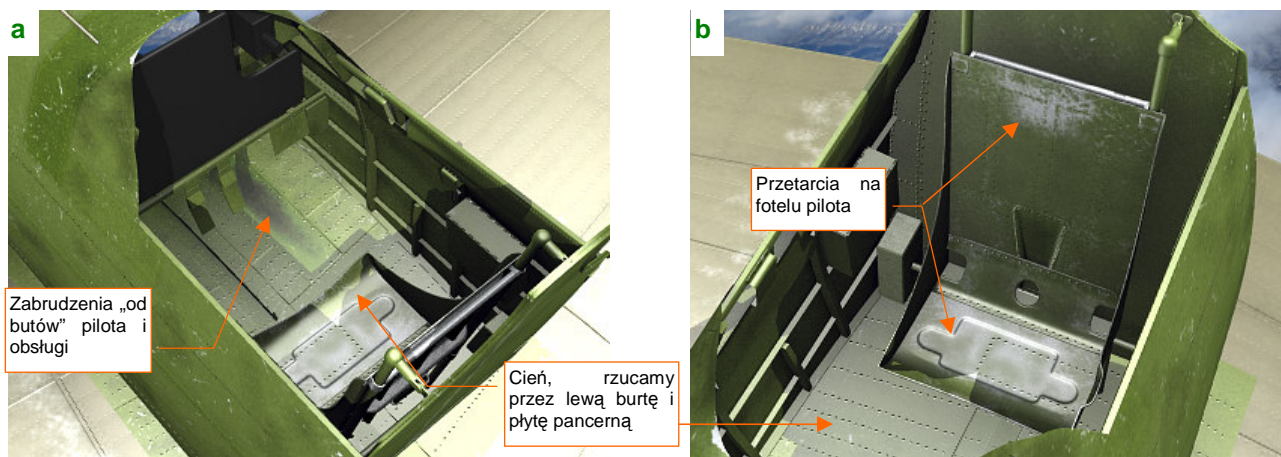
Rysunek 7.5.8 Tekstura barwy (*cockpit\color.png*)

Obraz rozwinięć UV wyeksportuj z Inkscape do pliku rastrowego, i wczytaj do Gimpa (jako plik *cockpit\cockpit.xcf*). Posługując się tymi samymi technikami, co w sekcji „Tekstura barwy” (str. 384) stwórz obraz tekstury barwy. Rezultat wyeksportuj do pliku *cockpit\color.png*.

Przygotowaliśmy w ten sposób trzy obrazy dla odpowiednich tekstur materiału **B.Skin.Cockpit**:

- plik *cockpit\nor.png* zawiera obraz dla tekstury nierówności **B.Cockpit.Nor**;
- plik *cockpit\ref.png* zawiera obraz dla tekstury rozpraszania światła **B.Cockpit.Ref**;
- plik *cockpit\color.png* zawiera obraz dla tekstury barwy **B.Cockpit.Color**;

Rysunek 7.5.9 przedstawia rezultat, osiągnięty na renderze po zastosowaniu tego materiału. (Dodałem do sceny półsferyczne światło od spodu, by lepiej było widać szczegóły „podłogi”):

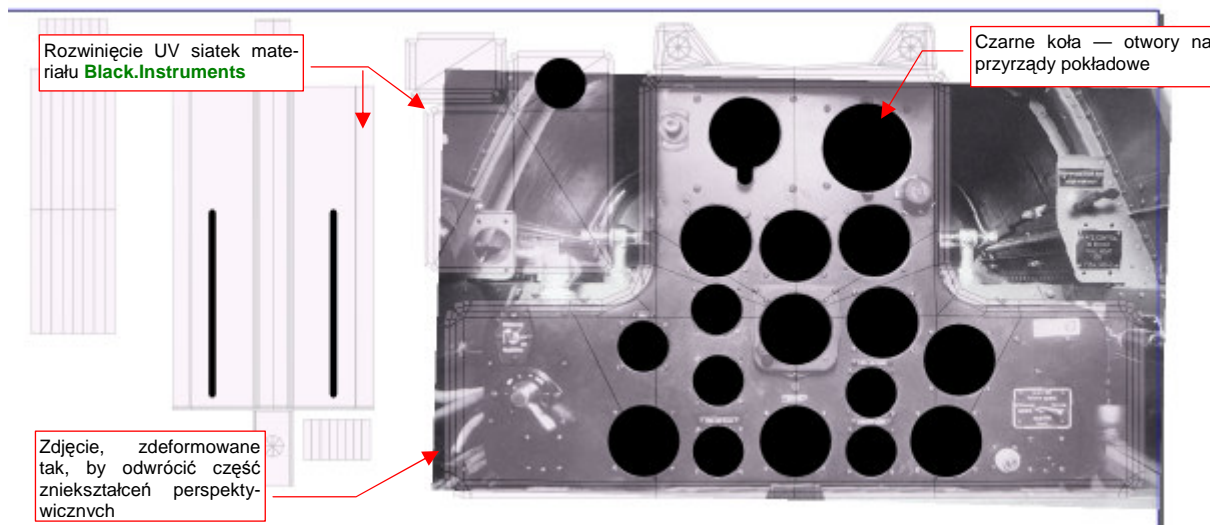


Rysunek 7.5.9 Rezultat zastosowania tekstur materiału **B.Skin.Cockpit**

Podłogą kabiny jest górna powierzchnia płata (Rysunek 7.5.9a). Obszary zakryte, tak jak ten, były nitowane za pomocą zwykłych nitów z łbami wypukłymi (a nie wpuszczanymi, jak na powłokach zewnętrznych). Jak przystało na podłogę, musi być zabrudzona, szczególnie tam, gdzie najczęściej stawiano stopy: w okolicach pedałów orczyka. Dodatkowo wzbogaciłem ją serią innych niewielkich zabrudzeń i zadrapań.

Fotel pilota (Rysunek 7.5.9b) należy zawsze do najbardziej „wytartych” elementów kabiny. Ten sam typ siedzenia był wykorzystywany w innym samolocie, produkowanym dla Wielkiej Brytanii: North American „Mustang”. Dokumentacja tego detalu to kilka zdjęć z restauracji P-51A, które znalazłem w Internecie. Wygląda na to, że o ile siedzenia pilota P-40 w USAAF pozostawiano w kolorze naturalnego duralu, to te fotele były malowane. W przypadku „Mustanga” były szaroniebieskie (farba **Dull Grey**)? Mogę tylko spekulować, w jakim kolorze były w samolotach AVG, więc pozostawiłem je w **Curtiss Green**, tak jak resztę kabiny.

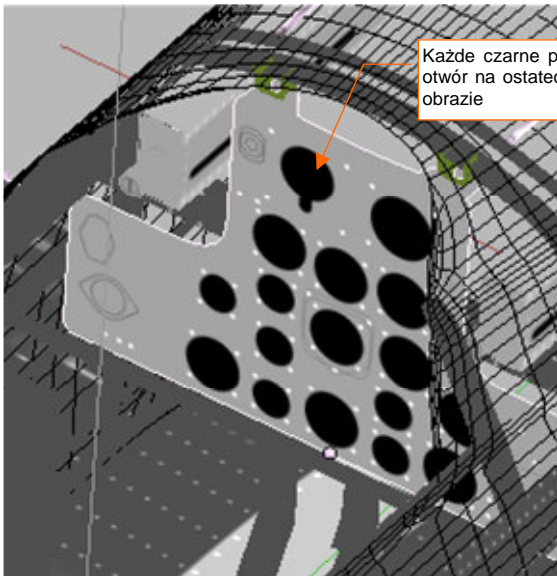
Tablica przyrządów jest najbardziej wyeksponowanym fragmentem kabiny pilota, więc warto jej poświęcić nieco więcej uwagi. Zaczniemy od naniesienia w Inkscape miejsc na tarcze przyrządów (Rysunek 7.5.10):



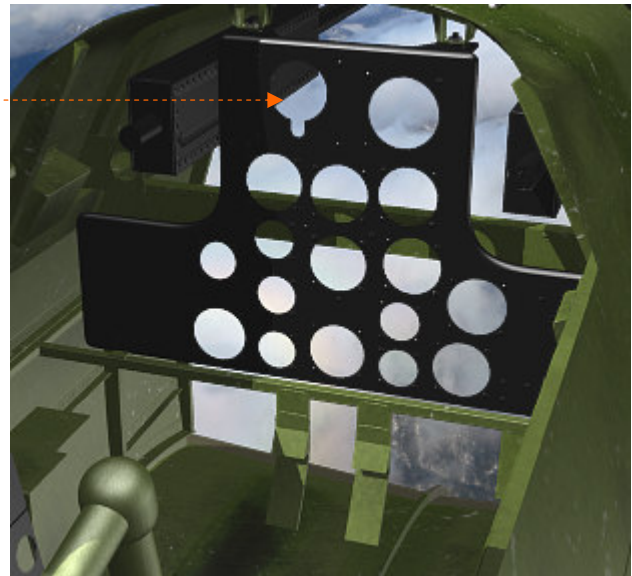
Rysunek 7.5.10 Przygotowanie tekstury otworów (głównie z myślą o tablicy przyrządów)

Pod rozwinięcia UV podstawiłem zdjęcie prawdziwej tablicy, i naniosłem w odpowiednich miejscach czarne koła.

Stworzony w Inkscape rysunek wyeksportuj do bitmapy. W trybie wyświetlania *Textured* podstaw go w Blenderze pod rozwinięcia siatek (jako plik przypisany do obrazu **Test-Instruments** — Rysunek 7.5.11). Sprawdź, czy wszystkie szczegóły dobrze „leżą” i czy nie są zdeformowane:



Każde czarne pole to otwór na ostatecznym obrazie

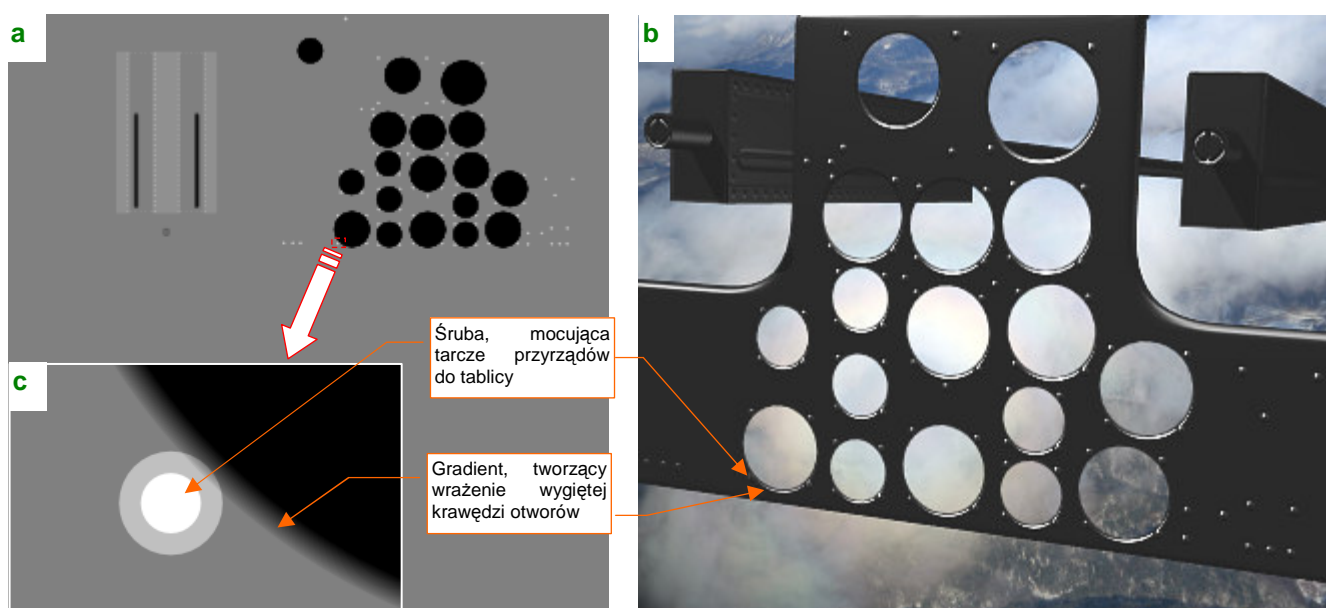


Rysunek 7.5.11 Dopasowanie obrazu otworów do rozwinięcia siatek (3D View, tryb *Textured*)

Rysunek 7.5.12 Test zastosowania tekstury otworów

Tarcze każdego wskaźnika, umieszczonego na tablicy, znajduje się w odpowiednim otworze. Przy krawędziach tych otworów widać okrągłe łby śrub, mocujących przyrządy. Śruby odwzorujemy za pomocą mapy nierówności. Otwory na tarcze przyrządów można byłoby zamodelować w siatce (wielu modelarzy tak robi). Ja jednak proponuję tu alternatywne rozwiązanie: zamiast wycinać dziur w siatce — uzyskamy je za pomocą tekstury nierówności **B.Instruments.Holes** (Rysunek 7.5.12). Efekt jest równie dobry, nawet przy użyciu obrazu o umiarkowanej rozdzielczości. (Użyłem pliku *instruments\holes.png*, który dla takiego rozłożenia siatek, jaki pokazuje Rysunek 7.5.3 ze str. 469, ma rozdzielczość 1024x1024 pikseli).

Błacha na krawędziach otworów w tablicy była wygięta do środka (typowy efekt technologii tzw. cięcia gumą). Aby odtworzyć ten szczegół, przygotowałem odpowiednią mapę nierówności (Rysunek 7.5.13a):



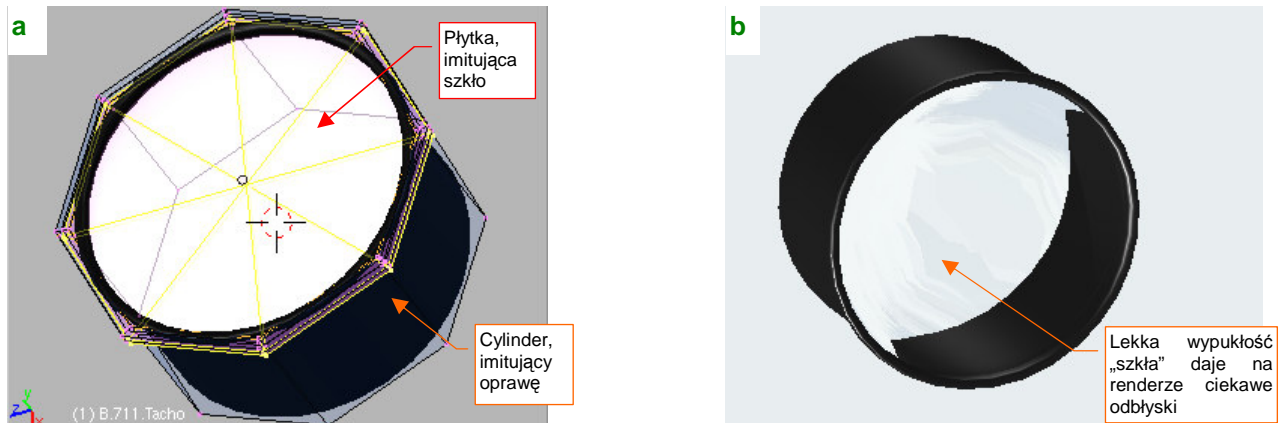
Śruba, mocująca tarcze przyrządów do tablicy

Gradient, tworzący wrażenie wygiętej krawędzi otworów

Rysunek 7.5.13 Szczegóły tekstury nierówności

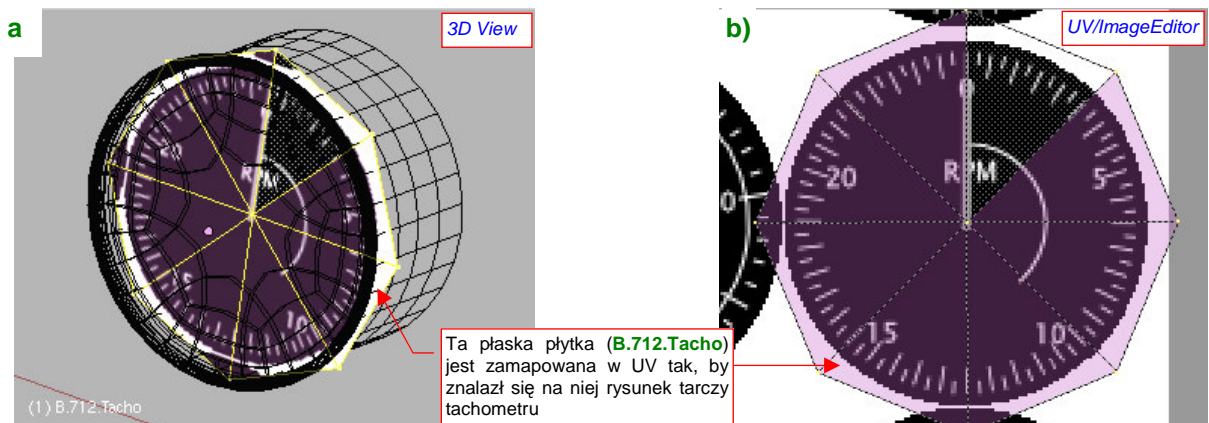
Otworów na obrazie (*instruments\nor.png*) tekstury nierówności są otoczone obrysem, wypełnionym odpowiednim gradientem kołowym (Rysunek 7.5.13c). Daje to całkiem poprawny rezultat (Rysunek 7.5.13b).

W tak przygotowane otwory wstawimy przyrządy. Większość z nich miała niemal identyczne oprawy (na tablicy występują w zasadzie tylko dwie średnice tarcz — wyjątkiem jest sztuczny horyzont). W tej sytuacji proponuję stworzyć większość opraw jako klony jednej siatki **B.711.Dial** (Rysunek 7.5.14):



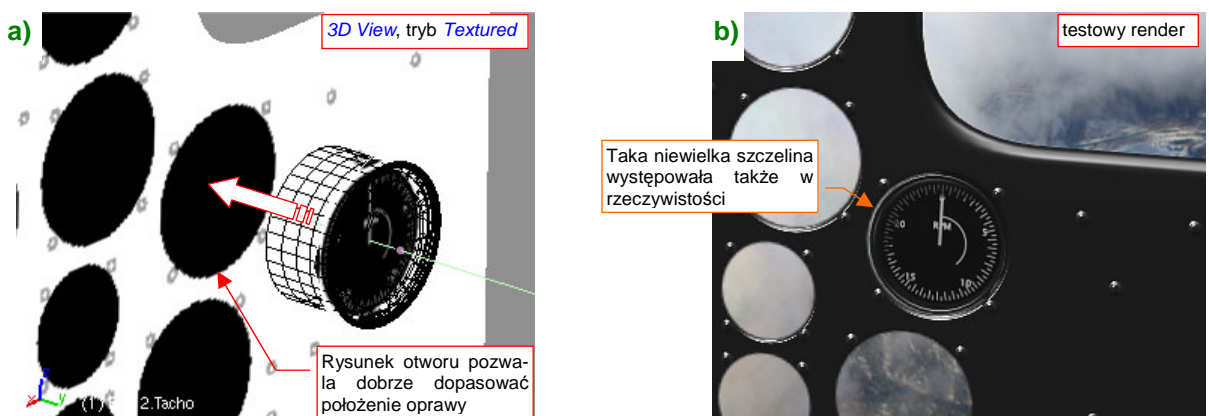
Rysunek 7.5.14 „Oprawa” wskaźnika

Oprawa składa się z cylindra, pokrytego tą samą czernią, co tablica przyrządów (**B.Instruments.Black**). W środek cylindra wstawiłem „płytkę”, imitującą szkło (Rysunek 7.5.14a). Ta płytka jest pokryta materiałem **B.Glass.Instruments** — kopią **B.Glass.Canopy**. Krawędzie „szkła” pozostawiłem zaokrąglone przez modyfikator *subsurf*, więc ich delikatna wypukłość pozostawia na renderze ciekawe odbłyski (Rysunek 7.5.14b). W środku oprawy umieściłem oddzielny obiekt. W tym przypadku to **B.712.Tacho** — płaska płytka, na której znajduje się rysunek tarczy przyrządu (Rysunek 7.5.15):



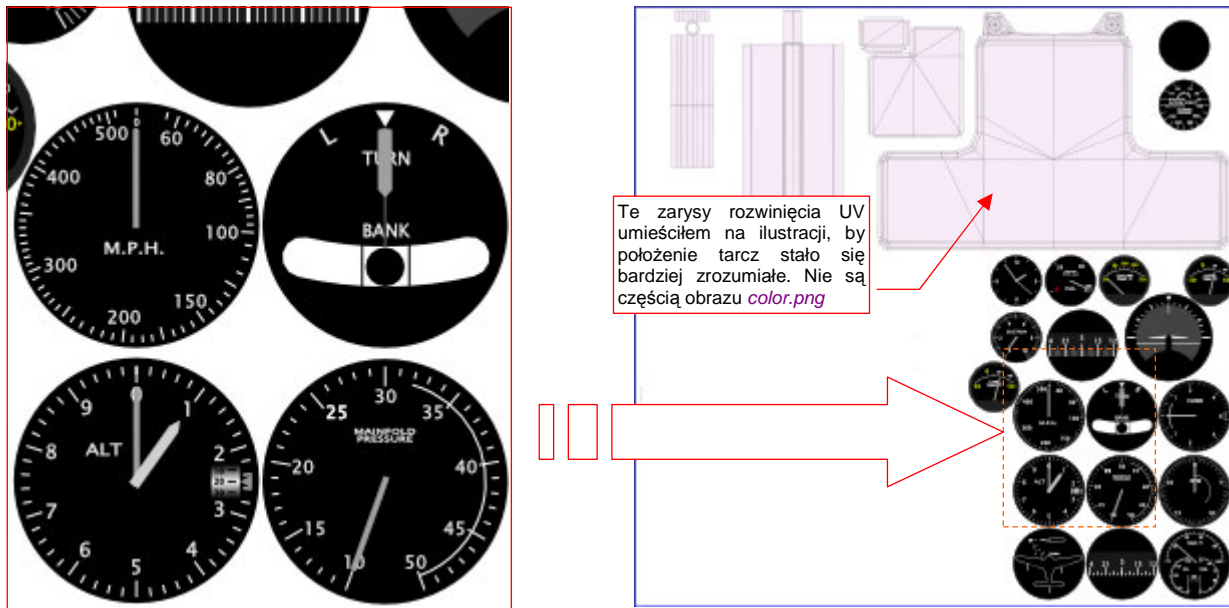
Rysunek 7.5.15 Tarcza wskaźnika

Potem pozostaje tylko umieścić oprawę w tablicy (najlepiej w trybie *Textured*, by trafić dobrze w otwór — Rysunek 7.5.16a). Rezultat na renderze pokazuje Rysunek 7.5.16b):



Rysunek 7.5.16 Osadzenie wskaźnika w tablicy

W ten sposób wykonaj wszystkie wskaźniki, umieszczone na tablicy przyrządów. Muszę przyznać, że najbardziej pracochłonną czynnością było ich narysowanie. (Wskazówki dot. szybkiego rysowania skali na tarczy przyrządu znajdziesz na str. 717). Stworzyłem je w Inkscape, na podstawie dostępnych zdjęć (Rysunek 7.5.17):



Rysunek 7.5.17 Tekstura B.Instruments.Color z narysowanymi tarczami wskaźników

Tarcze są dość duże — ich rozmiar dobrałem tak, by po eksporcie do obrazu rastrowego *instruments\color.png*, o wymiarach 1024x1024, nadal wyglądały w miarę dokładnie. Stworzenie takiego rysunku wymagało kilku godzin pracy. Rozdzielczość większości zdjęć kabiny, którymi dysponowałem, nie pozwalała na dokładne odczytanie cyfr i opisów na tarczach wskaźników. Czasami musiałem szukać podobnych przyrządów w konstrukcjach pokrewnych, np. w P-51. Rezultat wyszedł jednak dość zadowolająco (Rysunek 7.5.18):



Rysunek 7.5.18 Render tablicy przyrządów, po wstawieniu wszystkich wskaźników

Refleksy od lekko wypukłych szkieł przyrządów dodają całości realizmu. Paradoksalnie, w rzeczywistości szkło przyrządów było zupełnie płaskie. Jeżeli chcesz bardzo wiernie odtworzyć oryginał, możesz je „spłaszczyć”. Należy wówczas oznaczyć krawędzie obwodu „płytki”, którą jako zaznaczoną zaznaczył Rysunek 7.5.14a) (str. 473), jako ostre (*Cearse* = 1). Dzięki temu, że prawie wszystkie tarcze wykorzystują tę samą siatkę oprawy, wystarczy ją raz zmodyfikować — a wszystkie jej klony także ulegną zmianie.

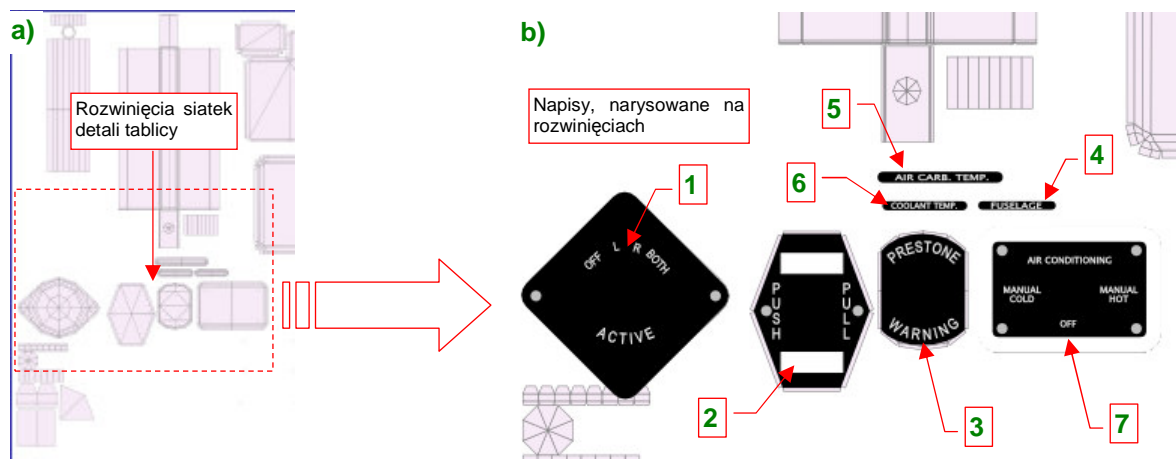


Teraz pozostaje uzupełnić tablicę przyrządów o kilka ostatnich detali (Rysunek 7.5.19):



Rysunek 7.5.19 Tablica przyrządów, uzupełniona o pozostałe szczegóły

Umieszczono na niej kilka plaketek z napisami (4, 5, 6, 7), wskaźnik ostrzegawczy przegrzania silnika (3), oraz parę przełączników (1, 2, 7). Siatki elementów, na których znajdował się tekst lub inny znak graficzny warto nanieść na teksturę barwy **B.Instruments.Color** (Rysunek 7.5.20a):

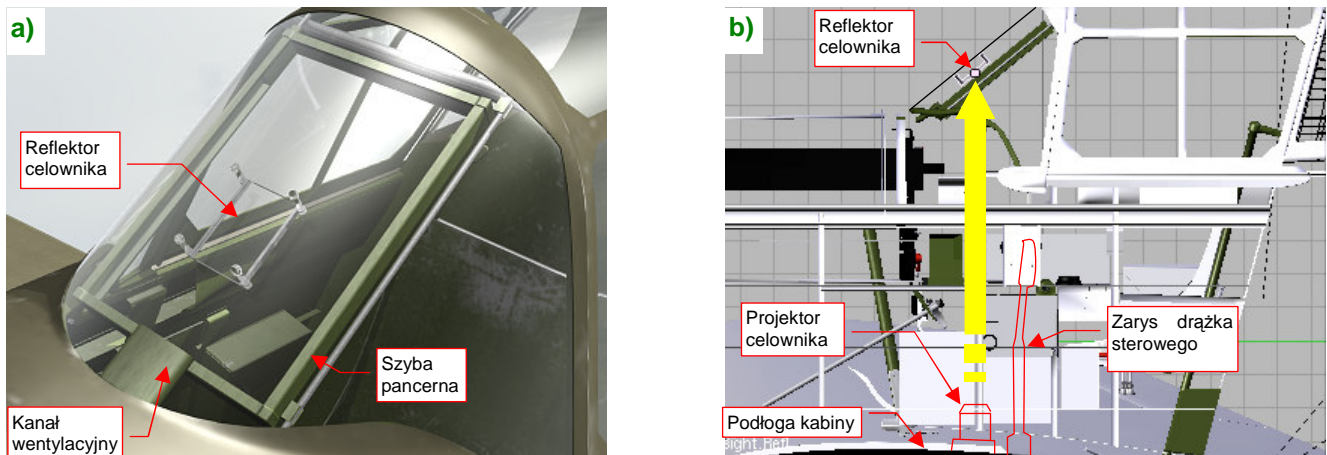


Rysunek 7.5.20 Napisy na detalach tablicy przyrządów — kolejny fragment obrazu [instruments\color.png](#)

(Nie będę tu opisywał tworzenia i rozwijania siatek, bo są trywialne — sądzę, że sam już dobrze wiesz, jak je wykonać). Rozwinięcia dodałem do warstwy UV w pliku [instruments\instruments.svg](#). Posługując się Inkscape, naniósłem na nie odpowiednie napisy i inne elementy (np. białe krawędzie). Na warstwie **Screws** dodałem także do grupy **#Screws** śruby, umieszczone w narożnikach plaketek. Po wygenerowaniu kolejnej wersji mapy kolorów ([instruments\color.png](#)) i nierówności ([instruments\nor.png](#)) uzyskałem efekt jak na renderze, który pokazuje Rysunek 7.5.19.

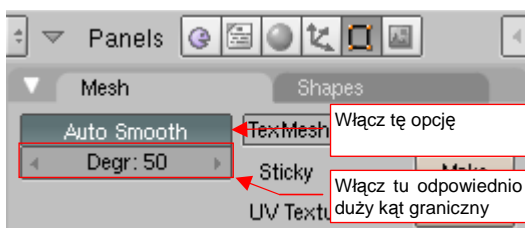
Zwróć może uwagę, że w niektórych elementach tablicy przyrządów wykorzystałem inne materiały, niż **B.Cockpit.Instruments**. Uchwyt przełącznika iskrowników (1) był czerwony, więc cały jest w materiale **B.Red.Details** (dzięki temu nie musiałem go rozwijać w UV). Podobnie przełączniki na plaketkach 2 i 7 to klony tej samej siatki, przypisane do materiału **B.Steel.Rough** (bo zapewnia im najbardziej odpowiedni „odblask”).

Tablica przyrządów jest już gotowa. To jednak dopiero „przygrzywka” do dalszej pracy nad wnętrzem kabiny. Czeka nas jeszcze wykonanie mnóstwa drobiazgów. Rysunek 7.5.21a) przedstawia wykończone wnętrze wiatrochronu:



Rysunek 7.5.21 Szczegóły wnętrza wiatrochronu

Pomiędzy prętami owiewki umieściłem ramkę z szybą pancerną. Szyba miała grubość 1.5 cala (38 mm). Nie byłem pewien, czy miała taki sam zielonkawy odcień, jak na P-40E. Przypisałem jej więc ten sam materiał „szkła”, jaki został użyty w pozostałych szymbach owiewki (**B.Glass.Canopy**).



Rysunek 7.5.22 Sposób na zaokrąglenia drobnych siatek — opcja **Auto Smooth**

zaokrąglania krawędzi — opcji **Auto Smooth**, z odpowiednio dużą wartością granicznego kąta pomiędzy ścianami (Rysunek 7.5.22).

W maszynach produkowanych dla USAAC ponad szybą pancerną był umieszczony reflektor celownika (Rysunek 7.5.21a). Projektor celownika była nietypowo umieszczona na podłodze kabiny (Rysunek 7.5.21b). To rozwiązanie odziedziczone po P-36. Samoloty produkowane dla Wielkiej Brytanii miały klasyczny, jednoczęściowy celownik refleksyjny za szybą pancerną.

Kłopoty mieli za to zbrojmistrze AVG, otrzymawszy brytyjskie samoloty bez celowników. Po wielu „podchodach” udało im się uzyskać amerykańskie projektory podłogowe (takie, jakie pokazuje Rysunek 7.5.21b). Musieli jednak sami do nich dorobić prowizoryczne reflektory (Rysunek 7.5.23).



Rysunek 7.5.23 Reflektor celownika, dorabiany dla samolotów AVG

Aby się w tych wszystkich szczegółach nie pogubić, w dalszej pracy nad kabiną proponuję przyjąć zasadę „wykańczania strefy”. Dzielisz w niej kabinę na kilka stref: wnętrze wiatrochronu, otoczenie tablicy przyrządów, lewą burtę, podłogę, prawą burtę. Każdą z nich wypełniasz wszystkimi detalami, i nie przechodzisz do następnej dopóty, dopóki aktualna nie będzie zupełnie wykończona. (Chodzi tu o pozostawienie wszystkiego — modelu, materiałów, tekstur — w takim stanie, byś nie musiał do nich nigdy wracać, by cokolwiek zmienić).

Pierwszą strefę — wnętrze wiatrochronu — już wykończyliśmy (por. Rysunek 7.5.21, str. 476), pora teraz uzupełnić detale wokół tablicy przyrządów (Rysunek 7.5.24):

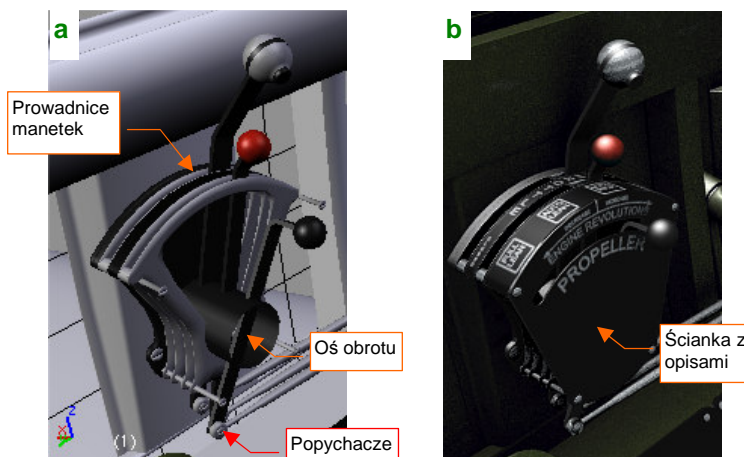


Rysunek 7.5.24 Detale wokół tablicy przyrządów

Najbardziej okazałym elementem tej strefy są pedały orczyka. Wnętrze kabiny wykonuję w sposób uproszczony, stąd zdecydowałem się użyć w obydwu tej samej siatki. W konsekwencji w przestrzeni **UV** materiału **B.Skin.Cockpit** istnieje tylko jedno rozwinięcie tego elementu. Wadą takiego podejścia jest identyczny układ rys i zabrudzeń na każdym z pedałów, ale nie rzuca się to specjalnie w oczy. Zaletą jest możliwość użycia przez siatkę większej powierzchni tekstury. Przyniosło to szczególnie dobry efekt w przypadku tekstury nierówności. Na renderze (Rysunek 7.5.24) na każdym z pedałów wygrawerowany napis eksploatacyjny („PUSH FORWARD TO RELEASE PARKING BRAKES”), oraz wyraźną, gęstą „kratkę” na końcach.

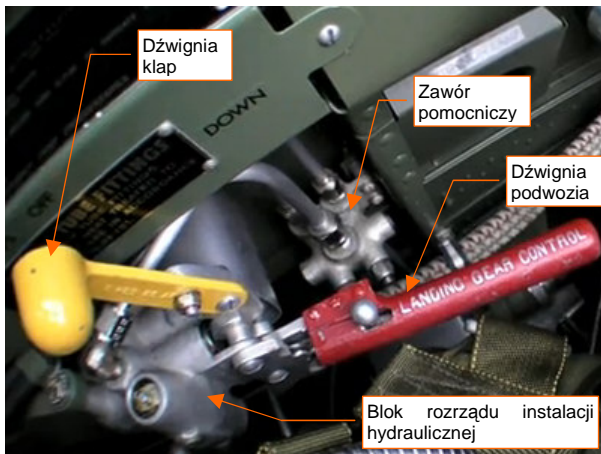
Oprócz orczyka odtworzyłem wiele innych detali — np. uchwyty służące do przeładowywania karabinów skrzydłowych, regulator ciśnienia tlenu, czy przełącznik sterujący odladzaniem gaźnika (Rysunek 7.5.24).

Po ukończeniu tej strefy, mamy gotowy cały przód kabiny, do wysokości tablicy przyrządów. Czas zabrać się za kolejny obszar — lewą burtę. Z tej strony znajduje się większość różnorodnych dźwigni i pokręteł. W szczególności zespół manetek silnika (przeputnicy i mieszanki) oraz skoku śmigła (Rysunek 7.5.25). Modelarze zazwyczaj odwzorowują obudowę tego zespołu jako trójkątne „pudełko”. Ze zdjęć wynika jednak, że była to raczej grupa blaszanych prowadnic (Rysunek 7.5.25a), bez jakichkolwiek ścianek bocznych. Z przodu była do niej tylko przykręcona „maskownica”, z napisami eksploatacyjnymi (Rysunek 7.5.25b).

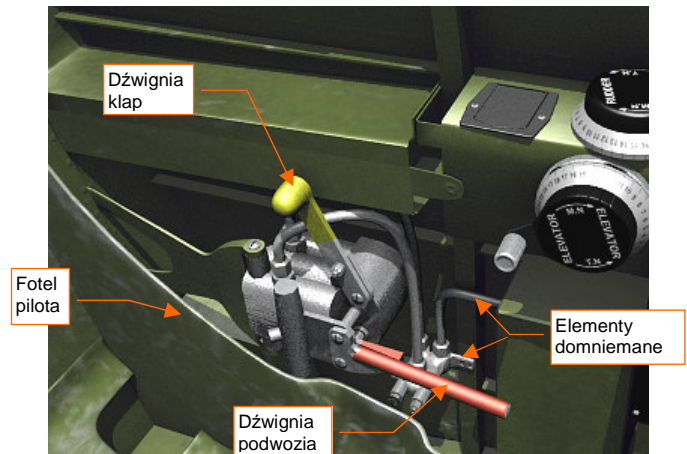


Rysunek 7.5.25 Szczegóły osłony manetek silnika i śmigła

Dalej wzdłuż lewej burty, przy fotelu pilota, znajdował się skomplikowany rozrząd instalacji hydraulicznej, z dwoma dźwigniami: klap i podwozia. Trudno jest znaleźć zdjęcia tego „wciśniętego” w kąt kabiny zespołu: wszystkie dokładniejsze fotografie pochodzą z późniejszych wersji P-40 (E i następnych) (Rysunek 7.5.26):



Rysunek 7.5.26 Zawór, sterujący klapami i podwoziem (P-40E)



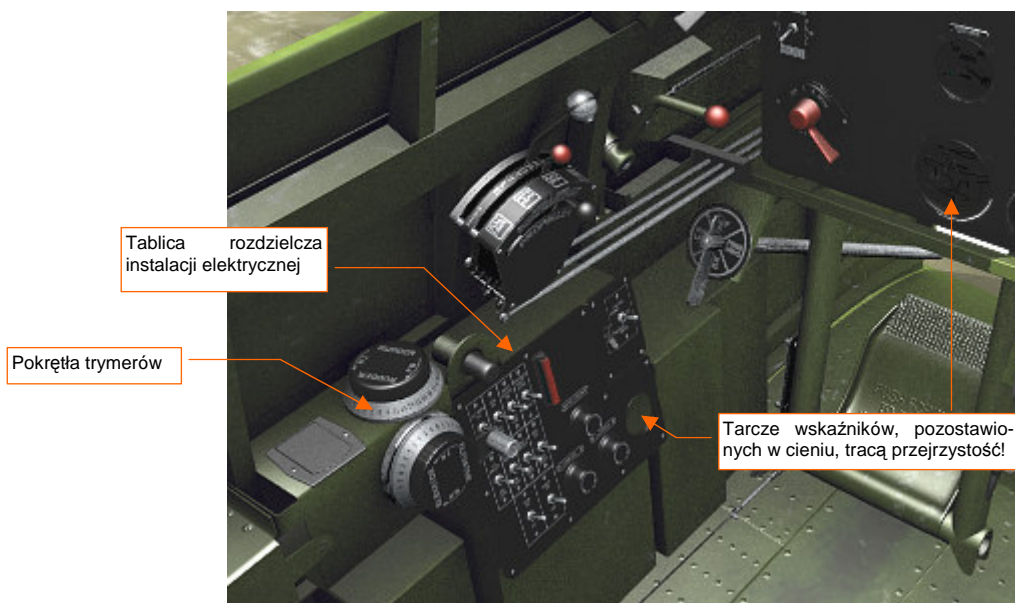
Rysunek 7.5.27 Odwzorowanie zaworu w naszym modelu

Tylko na dwóch zdjęciach pewnego P-40C w jednym z muzeów w USA dostrzegłem wystającą zza fotela dźwignię sterującą klapami skrzydłowymi. W dodatku ten egzemplarz nie ma całkiem oryginalnego wyposażenia: wygląda na to, że zespół sterowania silnikiem zaadaptowano z P-40E. Nie widać tam też śladu po przewodach, zasilających ten rozrząd. (Tak w ogóle, to w jego kabinie brak jeszcze wielu innych szczegółów). Nic dziwnego, że modelarze często pomijają ten detal, a ci, którzy się na niego natkną, czasami przypisują mu błędną funkcję. Spotkałem się np. z książką dla modelarzy ([14]), gdzie autor opisał ten zespół jako pompę paliwową!

Korpus rozrządu był na pewno odlewany — świadczą o tym obecne wszędzie zaokrąglenia i wyoblenia. Dokładne odtworzenie takiego kształtu byłoby bardzo pracochłonne. Ten szczegół jest tak mało widoczny, że zdecydowałem się odwzorować go w sposób uproszczony, bez zaokrągleń (Rysunek 7.5.27). Przebieg jednego z przewodów i sposób zamocowania pomocniczego zaworu są oparte wyłącznie na domysłach, które mogą okazać się fałszywe. Pomiąłem za to wiele innych przewodów, kryjących się za fotelem.

Kolejnym uproszczeniem jest także pominięcie napisów eksploatacyjnych, biegnących wzdłuż burt samolotu. Wprawdzie wykonanie napisów „DOWN” lub „OFF”, widocznych na fotografii (Rysunek 7.5.26) nie stanowiłoby większego problemu, ale co jest napisane na tej czarnej tabliczce obok?

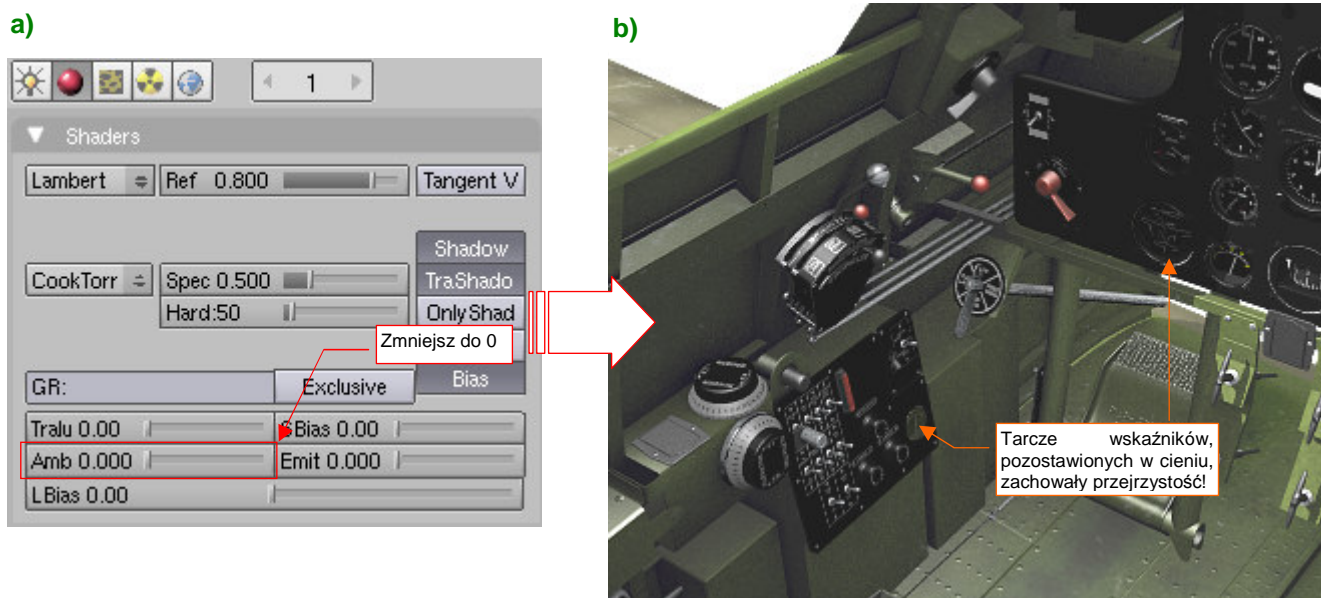
Innym szczegółem lewej burty jest tablica rozdzielcza instalacji elektrycznej (Rysunek 7.5.28):



Rysunek 7.5.28 Detale lewej burty (render z aktywnym efektem *Ambient Occlusion*).

Wykorzystałem na niej wiele klonów przełącznika, użytego po raz pierwszy na tablicy przyrządów (por. str. 475). Ponieważ lewa burta naszego modelu jest w cieniu, render, który przedstawia Rysunek 7.5.28 wykonałem z włączonym efektem *Ambient Occlusion* (por. 968). Pojawił się na nim brzydki efekt uboczny, czy może błąd Blendera: na tarczach wskaźników, znajdujących się w cieniu, nie widać niczego.

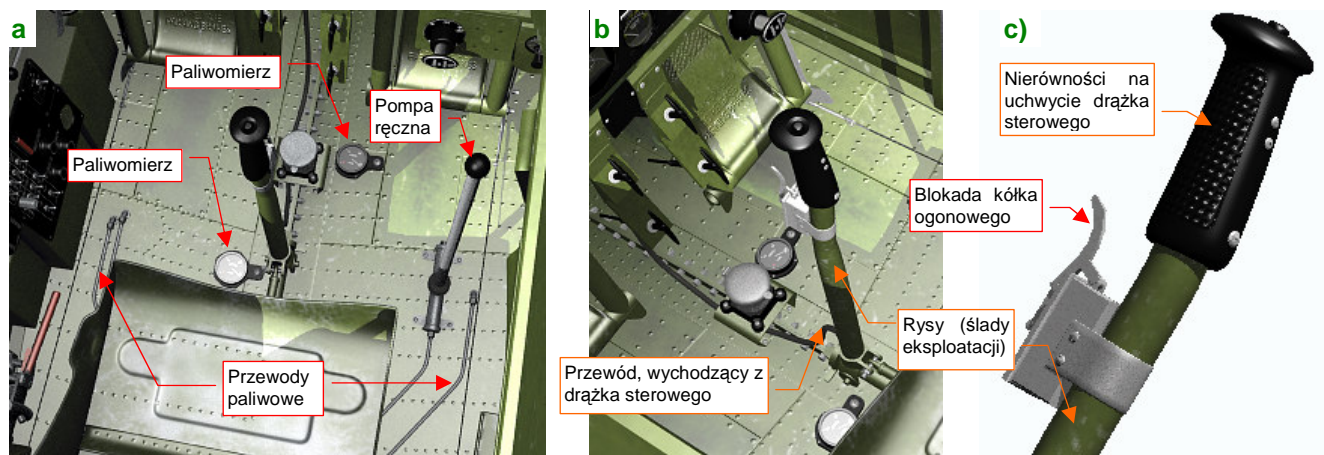
Ten błąd można obejść, minimalizując intensywność efektu *Ambient Occlusion* dla konkretnego materiału, w tym przypadku — instrumentów (**B.Black.Instruments**). Wystarczy zmniejszyć w panelu *Shaders* przełącznik **Amb** do 0.0 (Rysunek 7.5.29a):



Rysunek 7.5.29 Kompensacja błędu w obliczeniach *Ambient Occlusion*.

W wyniku zmniejszenia parametru **Amb**, tarcze zegarów przestały się chować w cieniu, i teraz pokazują swoje napisy. Taką samą zmianę należy wprowadzić w „pokrewne” czarne materiały, którymi są pokryte niektóre przełączniki i inne elementy kabiny: **B.Black.Opaque**, **B.Black.Details**.

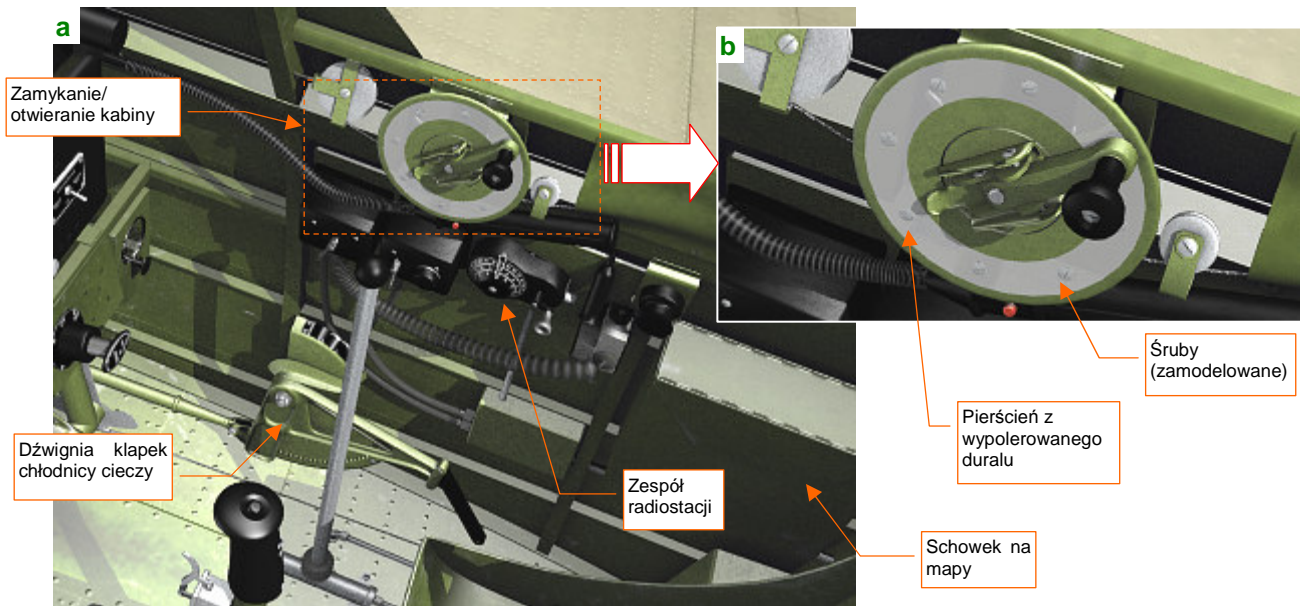
Kolejnym obszarem do wypełnienia jest podłoga kabiny (Rysunek 7.5.30). W P-40 tę funkcję pełniła górna powierzchnia płata. Na szczęście nie ma tu dużo detali do wykonania. Specyficzne dla tej konstrukcji są przymocowane do na podłogi paliwomierze zbiorników skrzydłowych (Rysunek 7.5.30a). Najważniejszymi elementami na tym obszarze są: zespół dźwążka sterowego i projektor celownika (Rysunek 7.5.30b):



Rysunek 7.5.30 Szczegóły podłogi kabiny pilota

Dźwążek był najbardziej narażony na rysy i zadrapania farby, więc rozwinąłem ten fragment siatki w przestrzeni **UV** materiału **B.Skin.Cockpit**. Zwróć uwagę na przewód, wychodzący z dźwążka. To szczegół dość często pomijany przez modelarzy. Rysunek 7.5.30c) przedstawia szczegóły górnej części dźwążka sterowego. Wytlóczenie na uchwycie uzyskałem za pomocą mapy nierówności (materiału **B.Black.Instruments**).

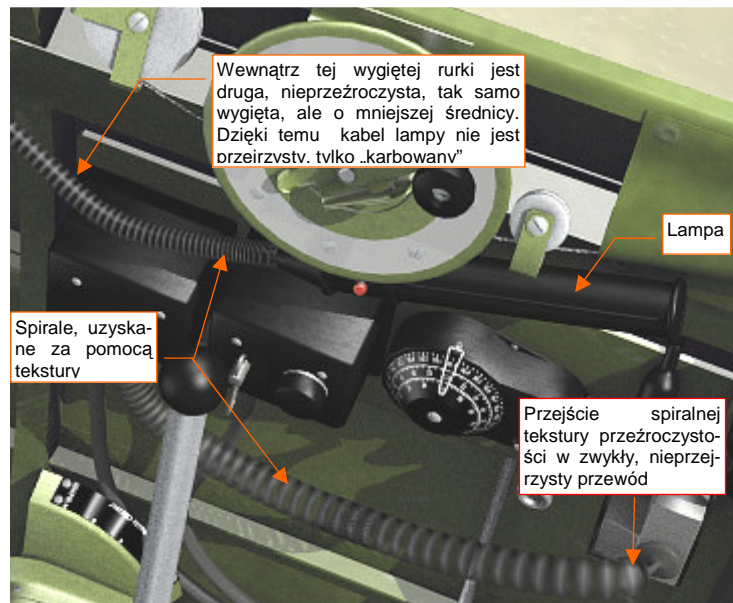
Wreszcie prawa burta kabiny (Rysunek 7.5.31). Konstruktorzy umieścili z tej strony „korbkę” służącą do odsuwania i zasuwania osłony, kontrolki radiostacji, oraz dźwignię sterowania chłodnicy (Rysunek 7.5.31a):



Rysunek 7.5.31 Szczegóły prawej burty kabiny pilota

Tarcza korby osłony kabiny miała fragment powierzchni zachowany w naturalnej barwie duralu, wypolerowanego niemal jak lustro (Rysunek 7.5.31b). Powinienem właściwie go pokryć materiałem **B.Skin.Duraluminum**, takim jak poszycie samolotu (por. str. 382). Przygotowanie dla tego tak małego fragmentu odpowiedniej tekstury nierówności byłoby jednak zbyt pracochłonne. Zrobiłem więc to „na skróty”: pokryłem tę powierzchnię materiałem, użytym do tłoków amortyzatorów podwozia (**B.Steel.Mirror**). Łby ośmiu śrub, które są na niej widoczne, to siatki, a nie tekstura (Rysunek 7.5.31b). (Wykorzystałem klony standardowych śrub, często używanych w innych miejscach kabiny i podwozia). Tak po prostu było najszybciej, a wpływ na komplikację modelu tych kilkudziesięciu dodatkowych ścian jest minimalny.

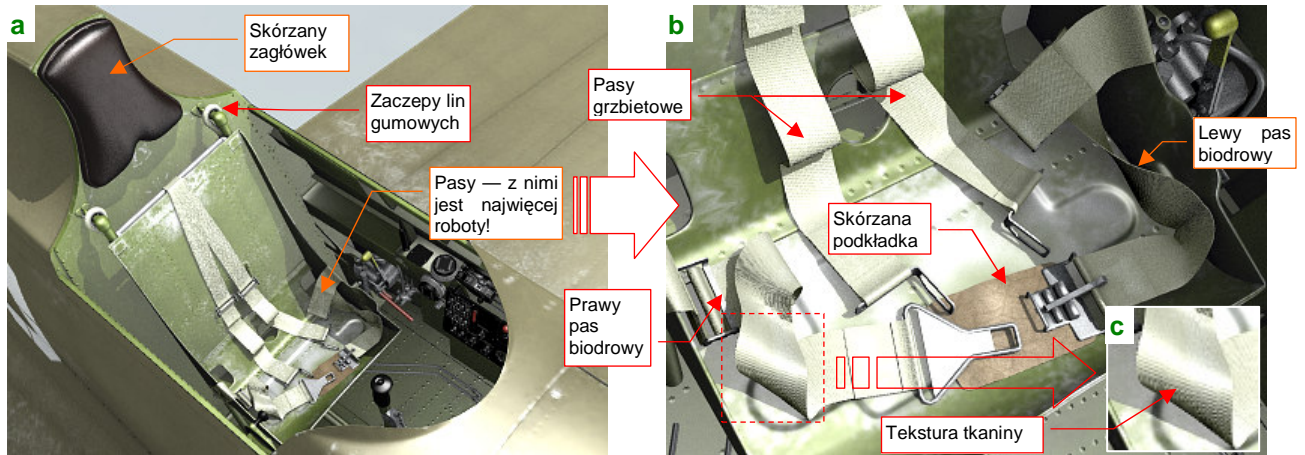
Na prawej burcie występują dwa spiralne przewody (Rysunek 7.5.32). Pierwszy to elastyczny kabel pomocniczej lampy (spotyka się takie nadal w niektórych lampkach biurowych). Drugi — to zawijany kabel od wtyczki (mikrofonu?) radiostacji, przypominający kable słuchawek starych, przewodowych telefonów. Są to na tyle drobne elementy, że zostały odwzorowane za pomocą tekstury. Już raz taką sztuczkę wykonaliśmy, przy sprężynach kółka ogonowego (por. str. 994). Wykorzystałem do tego celu, podobnie jak wtedy, odpowiednio spłaszczoną teksturę proceduralną typu **Wood**. Każdy z kabli to po prostu wygięty wzdłuż krzywej cylinder. Pewną niedogodnością jest tu konieczność indywidualnego dopasowania mapowania zwojów tekstury. Zwróć uwagę np. na zakończenie przewodu od radiostacji (Rysunek 7.5.32) — „kabel”, imitowany przez teksturę dyskretnie przekształca się w nieprzejrzystą końcówkę. Wymagało to „trafienia” końcem zwoju w początek nieprzejrzystego kabla. W rezultacie dla każdego z tych przewodów trzeba było stworzyć indywidualny materiał. Dla dwóch elementów, jak w tej kabine, to jeszcze nie był problem. Pamiętaj jednak o tym ograniczeniu, gdybyś chciał wykorzystywać to rozwiązanie częściej. Zbyt dużo materiałów może bardzo utrudnić wprowadzanie zmian w modelu.



Rysunek 7.5.32 Szczegóły spiralnych kabli

Wymagało to „trafienia” końcem zwoju w początek nieprzejrzystego kabla. W rezultacie dla każdego z tych przewodów trzeba było stworzyć indywidualny materiał. Dla dwóch elementów, jak w tej kabine, to jeszcze nie był problem. Pamiętaj jednak o tym ograniczeniu, gdybyś chciał wykorzystywać to rozwiązanie częściej. Zbyt dużo materiałów może bardzo utrudnić wprowadzanie zmian w modelu.

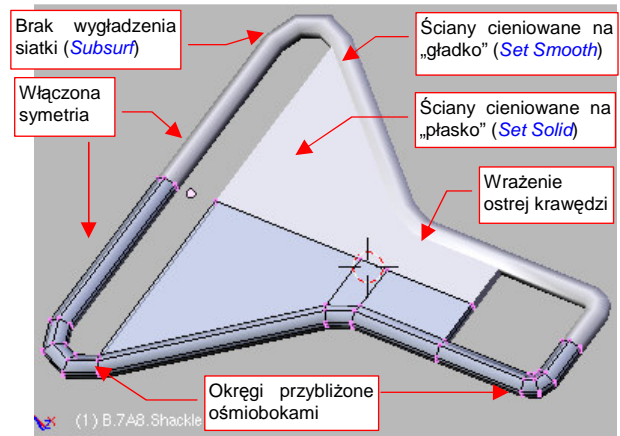
Ostatnim obszarem kabiny, który „uszczegółowiłem”, była tylna ściana, wraz z fotelem pilota (Rysunek 7.5.33):



Rysunek 7.5.33 Szczegóły tylnej ściany kabiny pilota

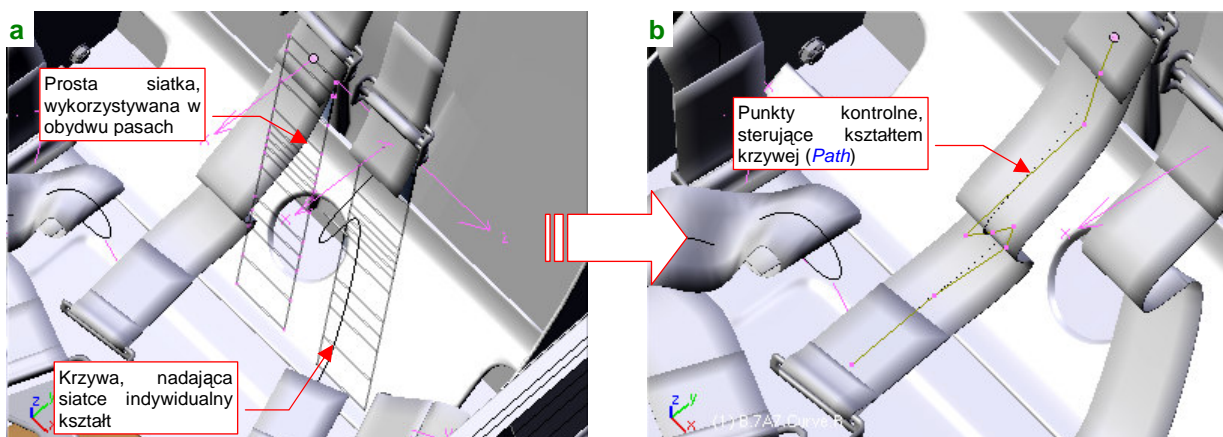
Wykonanie tego fragmentu wydaje się prostsze od pozostałych — bo zawiera mniej detali. Ot, zagłówek, pasy, i jakieś inne drobiazgi (Rysunek 7.5.33a). A jednak modelowanie „luźno rzuconej” uprząży pilota (Rysunek 7.5.33b) wymagało istotnego wysiłku. I to wcale nie z powodu zróżnicowanych materiałów!

Tworzenie tego zespołu zacząłem od wykonania kompletu sprzączek. Są to stosunkowo małe detale, więc nie trzeba było tu wygładzać siatki modyfikatorem *Subsurf*. Do uzyskania odpowiedniego efektu wystarczyło użyć uproszczeń — np. okręgu ośmiokątem (Rysunek 7.5.34). Wrażenie zaokrąglenia uzyskałem wyłącznie przez oznaczanie cieniowania odpowiednich ścian jako „gładkie” (*Set Smooth*). Ewentualne ostre krawędzie powstawały samoistnie na granicy ścian, różniących się tym ustawieniem. Zastosowałem także inne uproszczenia, jak lustrzane odbicie siatki. Rezultaty na renderze wyglądają całkiem realistycznie, nawet w zbliżeniach (Rysunek 7.5.33b).



Rysunek 7.5.34 Siatka sprzączki

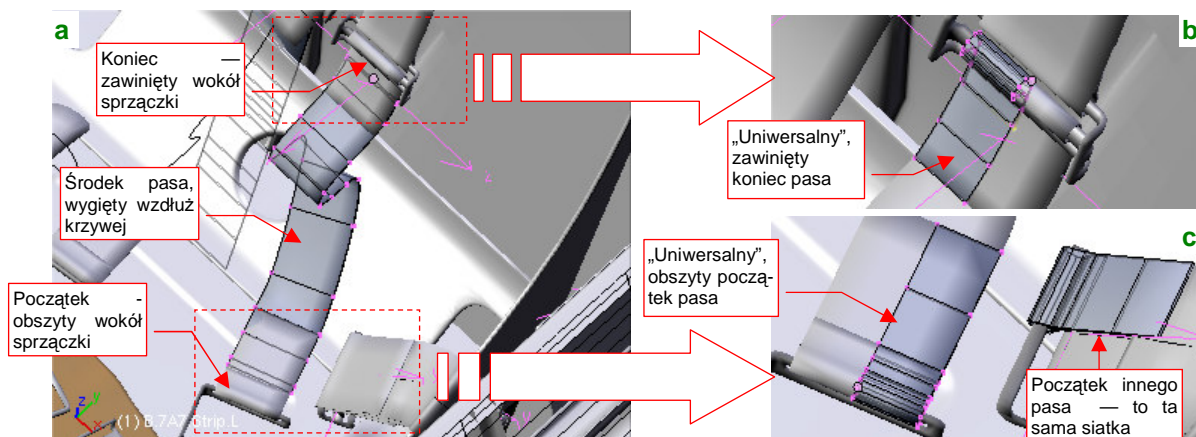
Same pasy to bardzo płaskie prostopadłości (Rysunek 7.5.35a). Ich grubość odpowiada 1 mm w rzeczywistym samolocie. Te siatki są wygięte wzdłuż odpowiedniej krzywej (modyfikatorem *Curve Deform*, tak jak wyginaliśmy kable i rurki). To te krzywe nadają im indywidualny kształt (Rysunek 7.5.35b):



Rysunek 7.5.35 Pasy — płaskie siatki, wygięte wzdłuż krzywej

Do stworzenia lewego i prawego pasa użyłem dwóch klonów tej samej siatki (por. Rysunek 7.5.35a).

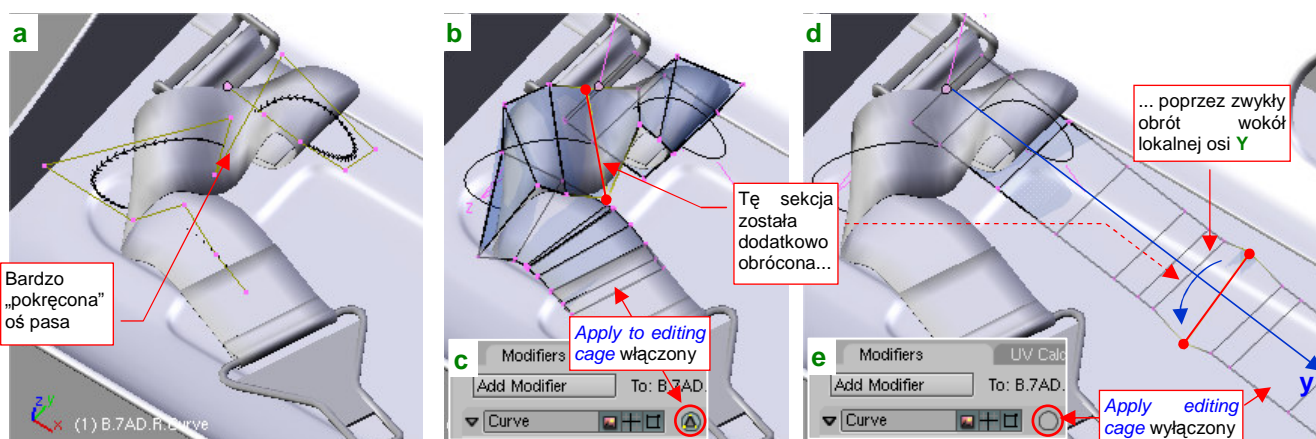
W zasadzie uprząż pilota składa się z czterech pasów: dwóch biodrowych i dwóch grzbietowych (choć te ostatnie są właściwie połączone w jeden długi odcinek — por. Rysunek 7.5.33a). W środku każdego pasa jest sprzączka, dzieląca go na dwie części. Początek każdej z tych części jest obszyty wokół jednej ze sprzączek, a koniec — zawinięty wokół poprzeczki drugiej (Rysunek 7.5.36a):



Rysunek 7.5.36 Trzy elementy każdego pasa: wygięty środek i dwie nie zdeformowane końcówki

Zamodelowanie takich końcówek poprzez wygięcie wokół krzywej jest możliwe, ale wymagałoby powtórzenia wiele razy tej samej, żmudnej roboty. (Mamy tu siedem fragmentów upręży, z których każdy ma swój indywidualny kształt — i swoją własną krzywą). Można jednak to zdecydowanie uprościć, gdy w charakterze obszycia użyjemy klonu jednej, uniwersalnej siatki. Dosztukujemy ją, jako oddzielny obiekt, do początku każdego pasa (Rysunek 7.5.36c). Podobnie każde owinięcie tkaniny wokół sprzączki to klon innej, uniwersalnej siatki, nakładany na koniec wygiętego odcinka pasa. Wystarczy tak skomponować krzywe, modelujące fragmenty upręży, by płaskie końcówki wydawały się być ich naturalnym przedłużeniem. (Obszycia, przez to, że łączą „na zakładkę” dwie warstwy tkaniny, rzeczywiście układają się płasko. Nie można tego powiedzieć o zawiniętych końcach pasów. Te jednak wystarczy pochylić pod różnymi kątami, by także nie raziły swoją „sztywnością”). Dzięki temu, że ani obszycie, ani zawinięcie nie są wyginane wzdłuż krzywej, można je jeszcze bardziej uprościć, modelując tylko połowę ich siatek — a drugą uzupełnić modyfikatorem *Mirror* (Rysunek 7.5.36b,c).

Przy bardziej „pokręconych” fragmentach pasów możesz mieć problemy z ich dopasowaniem do „sztywnych” końcówek. Dzieje się tak dlatego, że dla krzywych przestrzennych Blender sam ustala lokalny kąt skręcenia. Rysunek 7.5.37a) pokazuje przykład takiej sytuacji — prawy pas biodrowy, z bardzo wygiętą osią:



Rysunek 7.5.37 Korygowanie niewłaściwego skręcenia pasa wokół osi

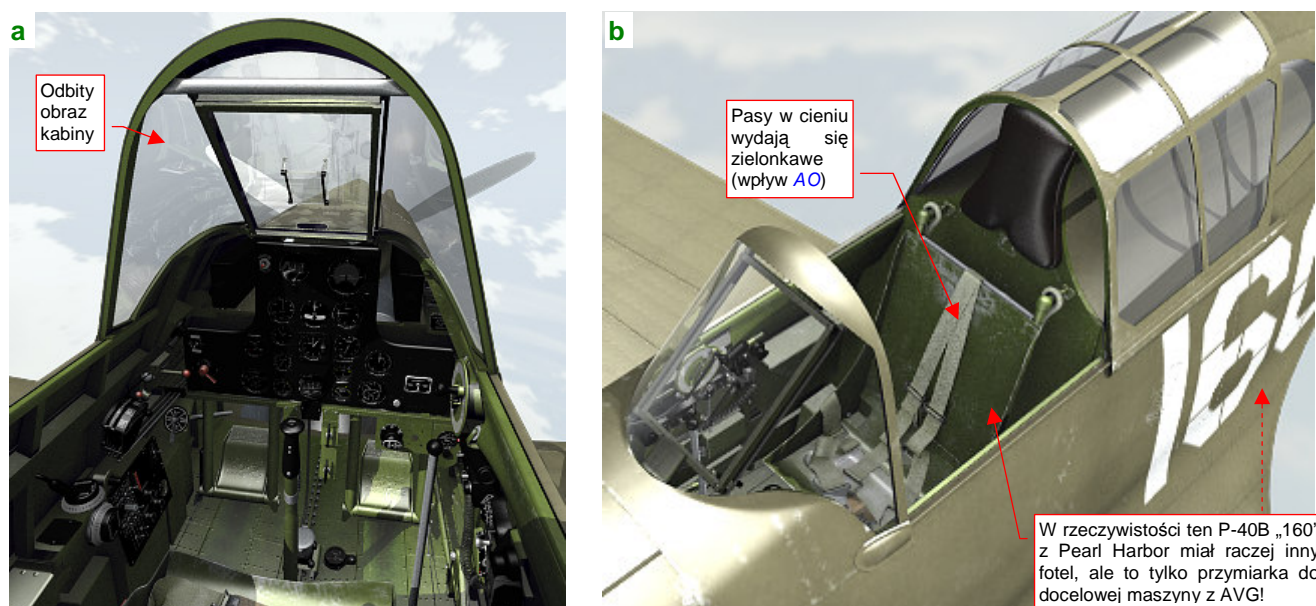
Udało się mi tu się tak dobrać kształt ścieżki, by „trafić” w sprzączki tego fragmentu. Jednak środek wymagał poprawy. Przy włączonym trybie *Apply to editing cage* (Rysunek 7.5.37c), można zaznaczyć niewłaściwie skręconą sekcję (Rysunek 7.5.37b). Potem wystarczy wyłączyć ten tryb (Rysunek 7.5.37e), i obrócić kłopotliwą krawędź wokół lokalnej osi **Y** (Rysunek 7.5.37d), por. także str. 918).



- Gdy zamierzasz skompensować automatyczne skręcenie siatki wokół krzywej, poprzez „ręczny” obrót jej krawędzi, stwórz najpierw jej lokalną kopię! Takie zmiany dotyczą zawsze konkretnego fragmentu uprząży, i nie powinny być powielane w pozostałych częściach (np. pasie lewym i pasie prawym)!

Mówiąc szczerze, układanie pasów uprząży tak, by wyglądały jak swobodnie rzucone na siedzenie, uważam za jeden z trudniejszych elementów modelowania samolotu. Gdy się z tym uporamy, nadchodzi pora, by przypisać poszczególnym obiektom odpowiednie materiały. Skórzanemu zagłówkowi przypisałem materiał **B.Black.Leather**, a podkładce pod „spinką” pasów (por. Rysunek 7.5.33b, str. 481) **B.Brown.Leather** (szczegóły konfiguracji - str. 1005). Pasom uprząży przypisałem materiał **B.White.Belt** (szczegóły konfiguracji — str. 1012), gdyż na zdjęciach i w filmach te pasy wyglądają na białe. Liny gumowe, zaczezione za fotelem, także wydawały się białe, więc przypisałem im ten sam materiał, co pasom (to pewne uproszczenie, ale nie razi).

Wnętrze naszej kabiny jest ukończone. Nie może się jeszcze równać z dobrymi symulatorami tego samolotu. Pomiąłem wiele przewodów i napisów eksploatacyjnych, których nie byłem w stanie odcyfrować. (Zawsze można je dodać później, gdy natkniesz się na nowe, dokładniejsze materiały). Tym niemniej osiągnęliśmy całkiem przyzwoite przybliżenie oryginału. Rysunek 7.5.38 przedstawia dwa ujęcia gotowej kabiny pilota:



Rysunek 7.5.38 Gotowa kabina (na razie bez pilota)

Rysunek 7.5.38a) to ujęcie (trochę ukośne) z wnętrza kadłuba. Zwróć uwagę na silne odbicia wnętrza kabiny na wygiętej szybie wiatrochronu. Co prawda ich intensywność zależy od punktu obserwacji. Gdy kamera znajduje się w osi samolotu i jest skierowana na wprost — pozostają widoczne głównie na szybie pancernej. Na wielu zdjęciach archiwalnych wnętrza kabin także można dostrzec zbliżone efekty<sup>1</sup>. Jeżeli uznasz je za nierealistyczne, możesz zmniejszyć w materiale **B.Glass.Canopy** ich wpływ — wystarczy przesunąć suwak *Ray Mir* w panelu *Mirror Transp* (por. str. 977).

Rysunek 7.5.38b) przedstawia kabinę widzianą z zewnątrz. W tym ujęciu źródło światła **Sun** jest nieco z tyłu, stąd cały fotel znalazł się w cieniu. Pasy wydają się na nim szarzielone — to wpływ efektu *Ambient Occlusion*. Tym niemniej i w tym stanie całkiem dobrze pasują do kabiny, więc nie zmniejszam im „czułości” na ten efekt (mam na myśli parametr *Amb* w panelu *Shaders* — por. str. 479).

<sup>1</sup> Swoją drogą, jeżeli szkło w rzeczywistym samolocie miało zbliżone właściwości, to nic dziwnego, że większość myśliwców z okresu II wojny światowej doczekała się wiatrochronów z płaskimi szybami. Taką zmianę owiewki przeszedł Spitfire (pomiędzy wersją V i IX), Messerschmitt (w wersji F), P-47 Thunderbolt, a także P-40. (Od wersji P-40D płaska płyta z kuloodpornego szkła stała się integralnym elementem centralnym nowego wiatrochronu. Boczne panele zachowały wygięcie, ale z szybą pancerną łączyły się pod kątem ostrym, minimalizując efekt odbicia otoczenia).

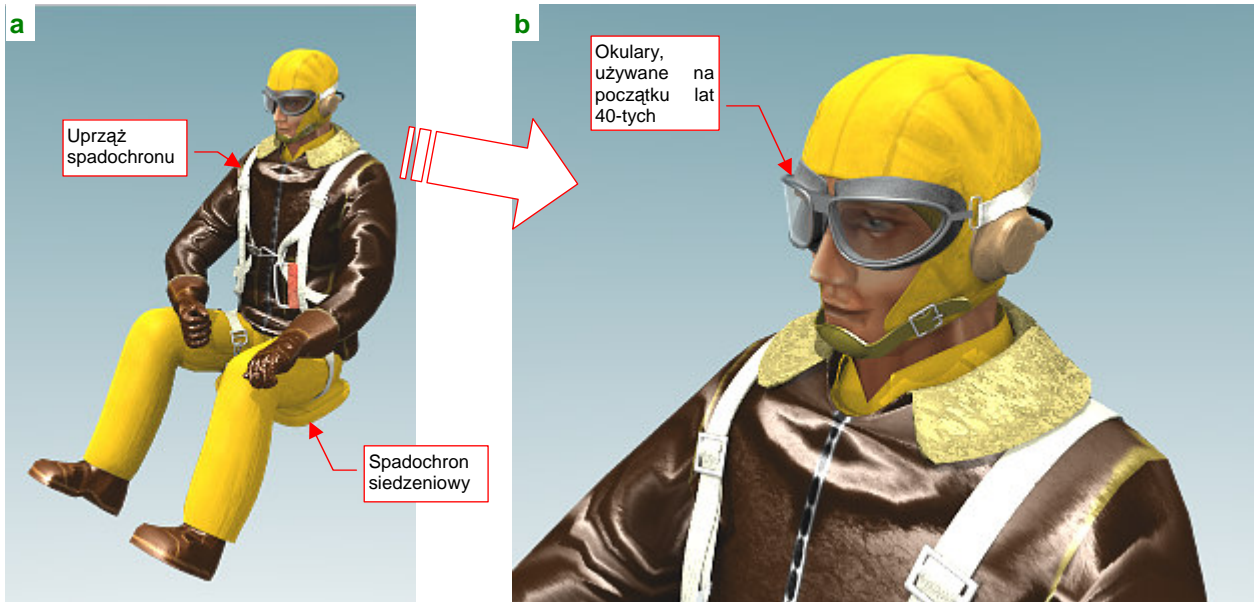
## Podsumownie

- Wnętrze kabiny samolotu odtwarza się głównie na podstawie zdjęć (str. 468, 478). Jest to zazwyczaj bardzo skomplikowany zespół. Dobrze jest zawsze przed rozpoczęciem pracy ustalić jakiś poziom uproszczeń — na przykład rezygnację z pomniejszych przewodów czy części napisów eksploatacyjnych;
- Rysunki w trzech rzutach przydają się głównie do dokładnego ustalenia położenia wręg i podłużnic wewnątrz kabiny. Te elementy szkieletu pełnią później rolę czegoś w rodzaju „układu współrzędnych”. Pomagają lepiej oszacować rozmiary i położenie poszczególnych elementów wyposażenia (str. 468);
- Pracę zaczynamy od identyfikacji i wymodelowania podstaw, do których są przymocowane wszystkie pozostałe elementy wyposażenia (str. 468). Sprowadza się to zazwyczaj do umieszczenia z przodu tablicy przyrządów, a jakiegoś zespołu „pudełek” po lewej i prawej burcie. Do „podstaw” zaliczam także fotel pilota (str. 469);
- W kabinie użyłem dwóch podstawowych materiałów (str. 468):
  - **B.Cockpit.Skin**: do „pomalowanych” powłok, czyli wewnętrznej strony poszycia samolotu, szkieletu, a także fotela pilota i innych większych elementów;
  - **B.Black.Instruments**: do wszystkich „czarnych” części: tablicy przyrządów, tabliczek z napisami eksploatacyjnymi;
 Dla każdego z tych materiałów trzeba przygotować rozwinięcia w **UV** odpowiednich siatek (str. 469). Posłużyłem się nim do mapowania tekstur nierówności (str. 470), odbić i barwy;
- Otwory w tablicy przyrządów najlepiej jest uzyskać za pomocą odpowiedniej tekstury przejrzystości (str. 471 - 472), oraz tekstury nierówności (str. 472);
- Każdy wskaźnik można wykonać jako zespół dwóch części:
  - szkła i oprawy, wspólnego dla wszystkich przyrządów. (Każdy ze wskaźników wykorzystuje klon tej siatki, w razie potrzeby — w zmienionej skali);
  - prostej tarczy zegara, która jest przygotowywana indywidualnie dla każdego wskaźnika (nanoszony jest na nią odpowiedni obraz wskazówek i skali);
 Tak przygotowany wskaźnik umieszczam w odpowiednim otworze tablicy przyrządów (str. 473 - 474);
- Efekt *Ambient Occlusion* (**AO**) potrafi zupełnie wymazać obraz umieszczonych „za szybką” wskaźników, gdy tylko pada na nie jakiś cień (str. 478). Obejściem problemu jest zmniejszenie „czułości” na efekt **AO** materiału, którym pokryte są tarcze — **B.Black.Instruments** (str. 479). Właściwie, to najlepiej jest wydzielić odrębny materiał wyłącznie dla tarcz przyrządów, aby nie ograniczać wpływu efektu **AO** na inne powierzchnie. Nie zdecydowałem się na ten krok wyłącznie dlatego, że nie chciałem bardziej komplikować tego przykładowego modelu;
- Tabliczki znamionowe i inne napisy eksploatacyjne to proste siatki, pokryte materiałem **B.Black.Instruments** (a właściwie jego „nieprzejrzystą” odmianą **B.Black.Opaque**). W ich rozwinięciach **UV** są umieszczone odpowiednie fragmenty tekstury barwy (str. 475);
- Szczegóły wyposażenia kabin mogą się różnić nawet w obrębie tej samej wersji samolotu. Widać to na przykładzie odmiennych reflektorów celowników w samolotach należących do USAAC i AVG (str. 476);
- Aby nie pogubić się w detalach, proponuję podzielić pracę nad kabiną na 5 stref:
  - przód, z tablicą przyrządów i wiatrochronem;
  - lewa burta;
  - podłoga;
  - prawa burta;
  - ściana tylna, wraz z fotelem pilota;
 Każdą z nich konsekwentnie wypełniamy wszystkimi szczegółami. Pracę nad następną strefą zaczynamy dopiero wtedy, gdy poprzednia została zupełnie skończona (str. 477);

- Dla różnych drobnych elementów, przygotowałem materiał **B.Inner.Details** (str. 475, 476). Efektywnie ma taką samą barwę, co wnętrze kabiny (**B.Skin.Cockpit**). Zawiera jednak tylko jedną, proceduralną teksturę „ogólnego zabrudzenia”, nałożoną w trybie *Orco*. Dzięki temu nie wymaga pracochłonnego rozwijania siatek w **UV**. Dla pokrycia detali w innych kolorach powstały z niego (zmieniłem tylko barwę podstawową) inne materiały: **B.Red.Details**, **B.Black.Details**;
- Wiele szczegółów trzeba wykonać w sposób uproszczony, o ile nie chcesz spędzić roku nad samą kabiną. Przykładem takiej części jest odlewany korpus sterowania hydrauliczną klapą i podwozia (str. 478). To niewielki, mało widoczny element, wciśnięty w kąt kabiny. Próba dokładnego odwzorowania jego kształtu pochłonięłaby tyle samo pracy, co całe skrzydło!
- Elastyczne, spiralne przewody można łatwo uzyskać za pomocą odpowiedniej tekstury przejrzystości (str. 480). Mankamentem tego rozwiązania jest to, że trzeba stworzyć odrębny materiał dla każdego rodzaju przewodu. Dobrze się za to nadaje się do wykonania kilku takich elementów w kabinie;
- Pasy fotela pilota najlepiej zbudować z odcinków, połączonych sprzączkami. Każdy odcinek pasa składa się z trzech elementów:
  - obszycia początkowej sprzączki (jedna, uniwersalna siatka, której klony są użyte w każdym odcinku);
  - środka (prostokątna siatka, wygięta wzdłuż krzywej w indywidualny kształt);
  - zawinięcia wokół sprzączki końcowej (uniwersalna siatka, której klony są użyte w każdym odcinku);Taka metoda pozwala uniknąć pracochłonnego „obszywania” każdej sprzączki (str. 481, 482);
- Blender automatycznie skręca pasy, wygięte wokół krzywej przestrzennej. Nieodpowiednie skręcenia można skompensować, skręcając w przeciwną stronę odpowiednie krawędzie siatki. Ośią obrotu powinna być zawsze lokalna oś **Y** „wykrzywionego” obiektu (str. 482);

## 7.6 Figura pilota

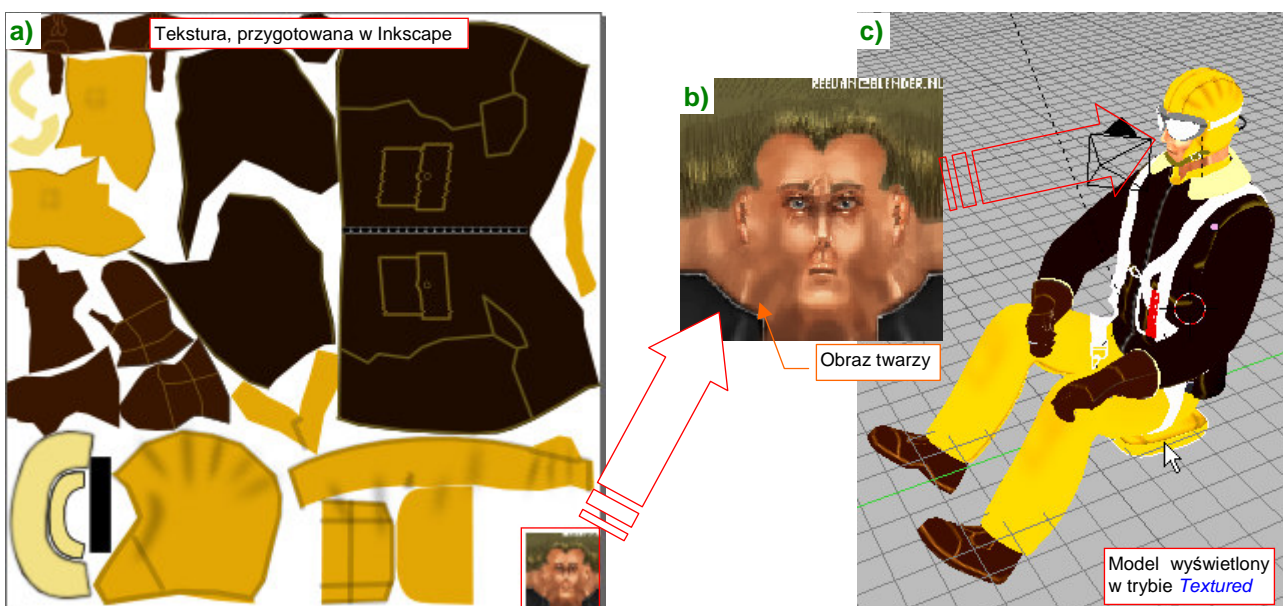
Kabina jest już gotowa, ale w scenach w locie dziwnie by wyglądała bez pilota. Tak się miło składa, że tę samą figurkę (byleby tylko była ubrana odpowiednio dla właściwych sił powietrznych) można wstawić do wielu modeli. Przygotowałem więc „uniwersalnego” pilota USAAC, w hełmofonie, goglach, kombinezonie i kurtce używanych w latach 1941-42. Założyłem mu także spadochron siedzeniowy, razem z uprzężą (Rysunek 7.6.1):



Rysunek 7.6.1 Figurka pilota USAAC/AVG (render)

Najbardziej szczegółowo przygotowałem górne partie tego modelu (Rysunek 7.6.1b) — gdyż te, po wsadzeniu pilota wabinę, będą najbardziej widoczne. Nie poświęciłem dużo uwagi na spodnie (np. pominąłem na nich kieszenie, znajdujące się pod kolanami) oraz buty (Rysunek 7.6.1a).

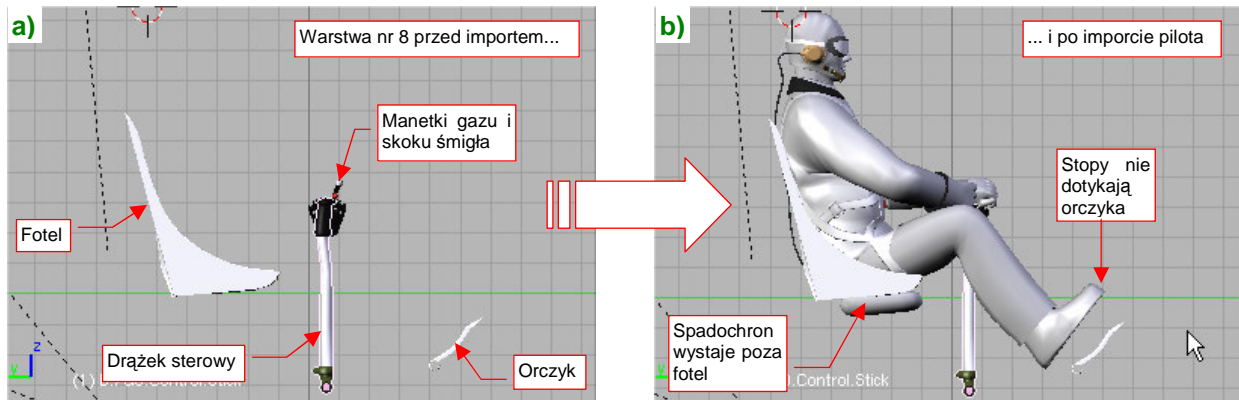
Plik z pilotem znajdziesz w folderze *source/pilot/*, pod nazwą *pilot(AVG).blend*. (Zdjęcia wykazują, że lotnicy AVG ubierali się tak samo, jak piloci USAAC). W pliku jest także „zapakowany” obraz tekstury, wykorzystywanej w modelu (Rysunek 7.6.2a). W prawym dolnym rogu umieściłem obraz twarzy (Rysunek 7.6.2b), zapożyczony z *Blender Art Magazine* (sam nie potrafiłbym tak namalować). Dla porządku zachowałem na nim podpis autora:



Rysunek 7.6.2 Tekstura i jej rozłożenie na modelu

Źródłowy plik obrazu (wykonany w Inkscape) tekstury znajdziesz w <p40/textures/7.06/pilot.src/000.svg>.

Jeżeli sprawdzisz wcześniej sam plik *pilot(AVG).blend* w Blenderze, przekonasz się, że pilot znajduje się w nim na warstwie 8, niewidoczne elementy pomocnicze (krzywe) na 9, a armatura („kości” — por. str. 815) — na warstwie 10. Na tych samych warstwach pojawiają się po wczytaniu do modelu P-40. Warto więc włączyć elementom, do których pilot musi się „dopasować” — dodatkową obecność na warstwie 8 (Rysunek 7.6.3a). Potem wystarczy wczytać z pliku *pilot(AVG).blend* figurkę (poleceniem **File → Append or Link**, szczegóły str. 767) :

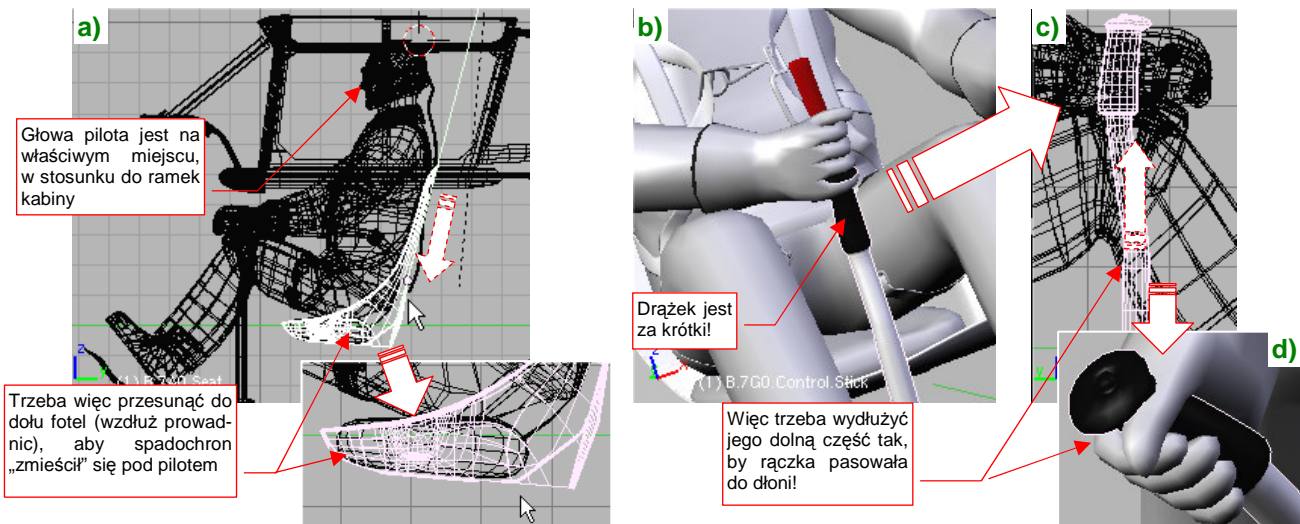


Rysunek 7.6.3 Import pilota do naszego modelu

Rysunek 7.6.3b) pokazuje sytuację zaraz po wczytaniu: zazwyczaj, trzeba dokonać kilku dopasowań. Nie inaczej jest w tym przypadku: spadochron wystaje za dno fotela, a stopy nie sięgają orczyka. Dodatkowo włączyłem widoczność ramek kabiny (warstwa 4) (Rysunek 7.6.4a). Wygląda na to, że głowa pilota znajduje się na prawidłowym poziomie, więc trzeba obniżyć fotel, by dopasować go do spadochronu.

- Gdybyś miał chcieć przesunąć pilota, „łap” go za pakunek spadochronu (**000.Parachute**). To do niego są przypisane wszystkie pozostałe elementy (także te na innych warstwach: krzywe i armatura) .

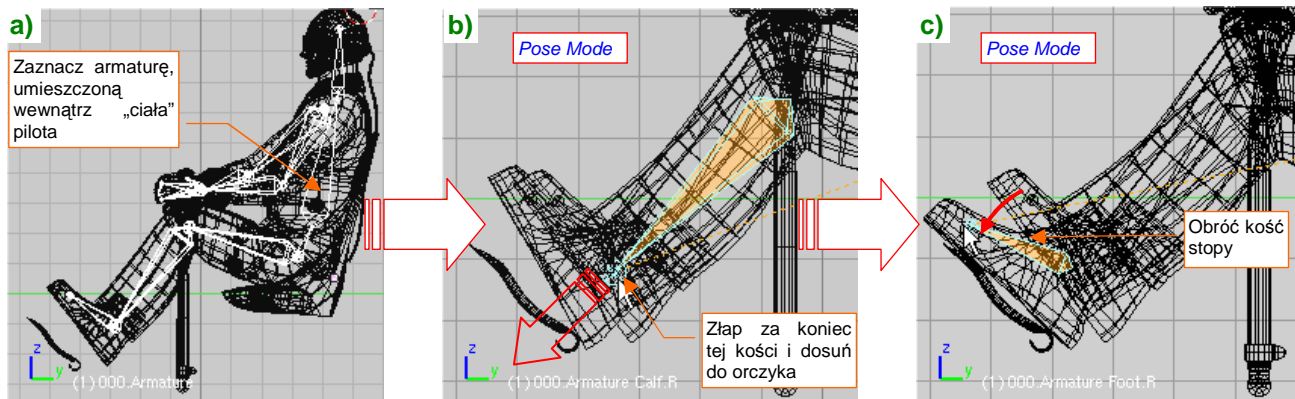
Rysunek 7.6.4a) przedstawia dopasowywanie do „poziomu siedzenia” pilota. Tak się dobrze składa, że lokalna oś **Y** fotela biegnie wzdłuż prowadnic — więc wszystko, co trzeba zrobić, to przesunąć go do dołu wzdłuż tej osi. Wystarczy obserwować (w trybie *wireframe*), kiedy spód siedzenia pilota obejmie cały spadochron:



Rysunek 7.6.4 Dopasowanie elementów kabiny do pilota

Teraz przyjrzyj się uważnie rękom pilota. Lewa pewnie spoczywa na manetkach gazu i obrotów — nie wymaga żadnych poprawek. Za to z prawą jest kłopot: drążek sterowy jest za krótki o jakieś 10 cm (licząc w jednostkach rzeczywistych) (Rysunek 7.6.4b). To raczej wygląda na mój błąd podczas modelowania — nieprawidłowo oceniłem długość tego elementu na podstawie zdjęć. Na szczęście nie jest to jakiś duży problem: wydłużyłem siatkę w dolnej części drążka (tam, gdzie była prostą rurą) (Rysunek 7.6.4c). Zrobiłem to tak, by zachować proporcje pomiędzy częścią prostą i wygiętą. Potem pozostało jeszcze przesunąć do góry parę elementów związanych z rączką, i całość znalazła się na właściwym miejscu (Rysunek 7.6.4d).

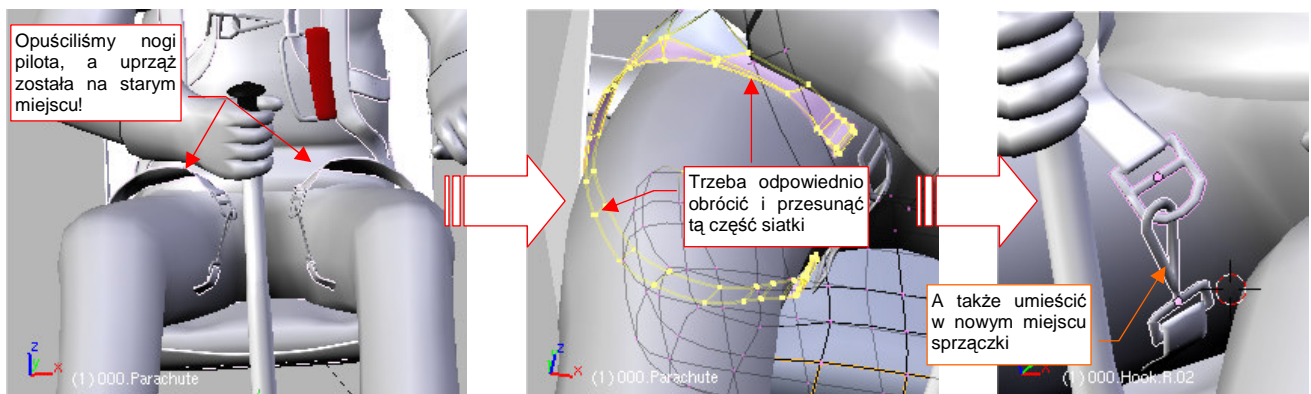
Teraz trzeba wyprostować nogi pilota, by spoczęły na pedałach orczyka. Model, który wczytałeś, ma siatkę przypisaną do armatury („kości”). Wystarczy włączyć dodatkowo widoczność warstwy 10, na której je umieściłem. W związku z tym, że armatura znajduje się wewnątrz „ciała” lotnika, najwygodniej ją zaznaczyć po przełączeniu trybu wyświetlania na *wireframe* (Rysunek 7.6.5a):



Rysunek 7.6.5 Dostosowanie stóp pilota do pedałów orczyka

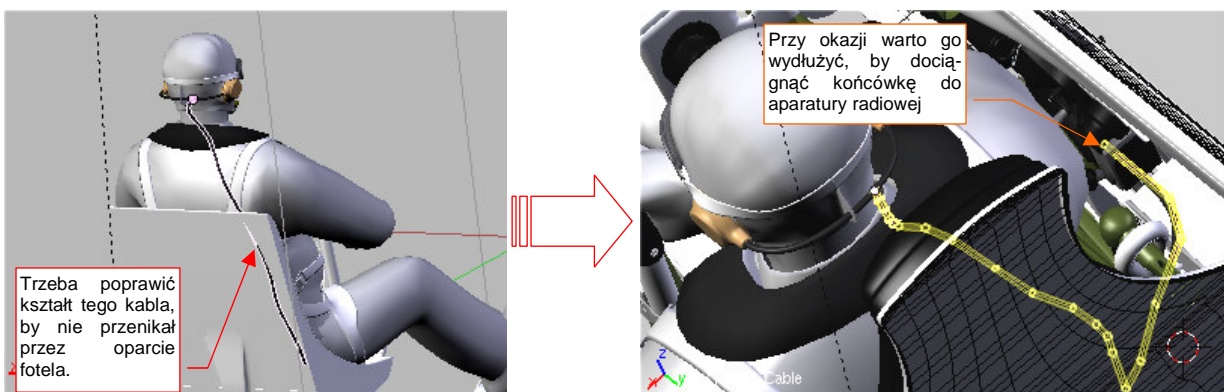
Następnie przełącz się w *Pose Mode* (por. str. 817), i zaznacz koniec kości łydki (Rysunek 7.6.5b). Przesuń ją tak, by obcas oparł się o pedał. Następnie zaznacz kość stopy (Rysunek 7.6.5c), i obróć ją, by cała podeszwa buta spoczęła na orczyku. Tak samo zmień położenie drugiej nogi.

W rezultacie poprawnego ustawienia nóg trzeba poprawić pasy uprząży na udach pilota (Rysunek 7.6.6). (Zawsze zapominam przypisać ją do kości armatury).



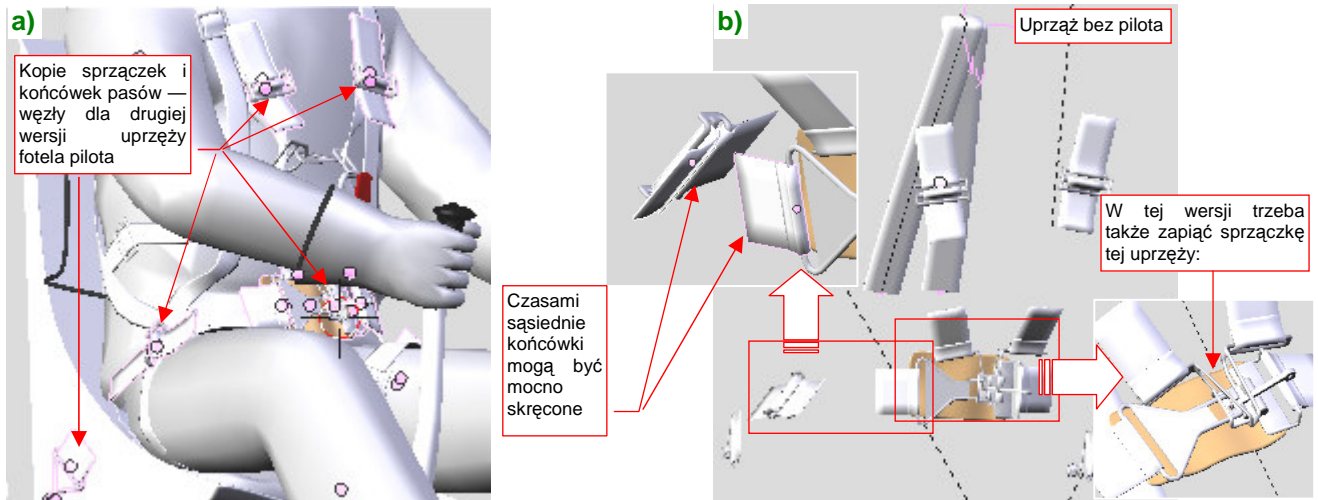
Rysunek 7.6.6 Poprawianie uprząży spadochronu

Trzeba przejść do edycji siatki uprząży spadochronu, i odpowiednio obrócić — tak, jak obróciły się uda — odpowiednie fragmenty siatki. Potem należy dopasować sprzączki do zmienionego położenia końcówek pasów. Ostatnim elementem, który na pewno będzie wymagał poprawienia, jest kabel radiowy, biegnący z hełmofonu pilota (Rysunek 7.6.7). Trzeba go inaczej wygiąć, by nie przechodził przez oparcie fotela, i wydłużyć:



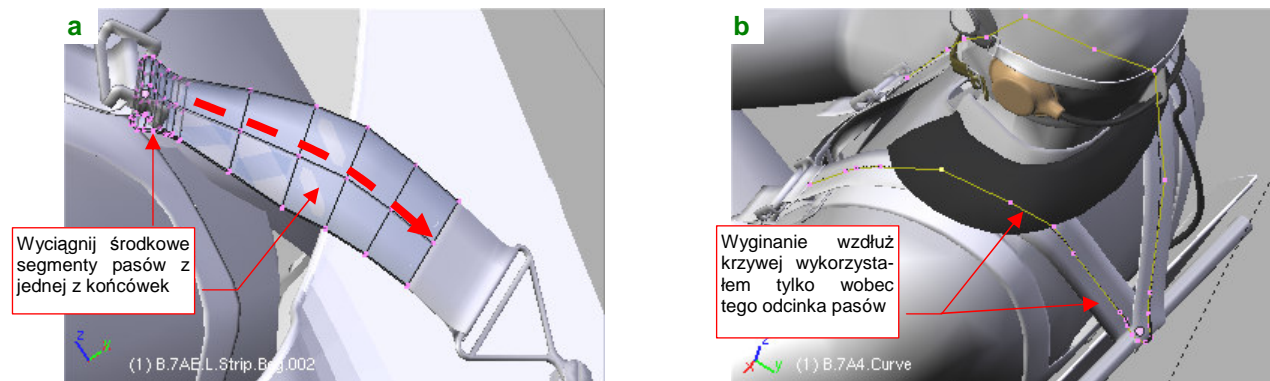
Rysunek 7.6.7 Poprawianie przebiegu kabla hełmofonu

Pozostaje „przywiązać” pilota do fotela. Umieść w tym celu na warstwie 8 wszystkie sprzączki i końcówki pasów upręży, którą przygotowaliśmy w poprzedniej sekcji (por. str. 481). Rozmieść je w nowych położeniach — tam, gdzie powinny się znaleźć na ciele pilota (Rysunek 7.6.8a). To drugi zestaw pasów, do scen w locie:



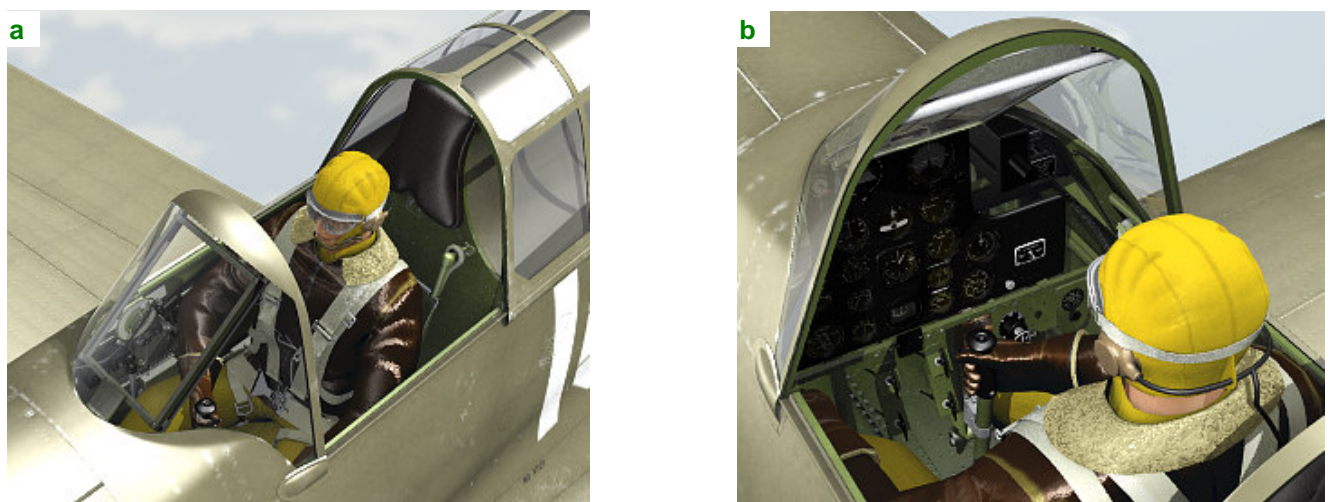
Rysunek 7.6.8 Ponowne rozłożenie upręży fotela pilota

Przy okazji trzeba „zapiąć” wszystkie elementy sprzączki (Rysunek 7.6.8b), i ustawić odpowiadające sobie węzły tak, by mimo skrócenia miały szansę zostać poprawnie połączone. Tym razem pasy nie są tak bardzo „pokręcone”, więc najlepiej jest je „wyciągnąć” z jednej z końcówek (Rysunek 7.6.9a). Krzywej użyłem tylko dla najdłuższego odcinka na plecach (Rysunek 7.6.9b):



Rysunek 7.6.9 Szczegóły wykonania pasów

Rysunek 7.6.10 przedstawia pilota w ujęciu z przodu (a) i z tyłu (b). „Słońce” świeci tu z tylnej półsfery.



Rysunek 7.6.10 Pilot w kabinie — ujęcia z przodu i z tyłu

Rysunek 7.6.11 przedstawia bardziej szczegółową wersję ujęcia z przodu. Tym razem umieściłem słońce na przedniej półsferze, stąd odbłask na wiatrochronie. Przy takim, bardziej intensywnym, oświetleniu, zdecydowałem się odrobinę przesunąć teksturę „fałd” na kurtce pilota, uzyskując intensywne odbłyski (por. str. 1011):



Rysunek 7.6.11 Ukończona kabina, wraz z pilotem

Uważam, że pilot i kabina na tym testowym renderze wyglądają zupełnie dobrze. Co prawda, dla takiego ujęcia warto by było zastosować na zewnętrznych powierzchniach samolotu teksturę barw o większej rozdzielczości. Taką decyzję trzeba było jednak podjąć wcześniej — teraz obraz poszycia jest już wykonaną w GIMP, i trzeba byłoby go tworzyć od nowa. Tak więc pamiętaj o tym na przyszłość, jeżeli planujesz zbliżenia!

### Podsumownie

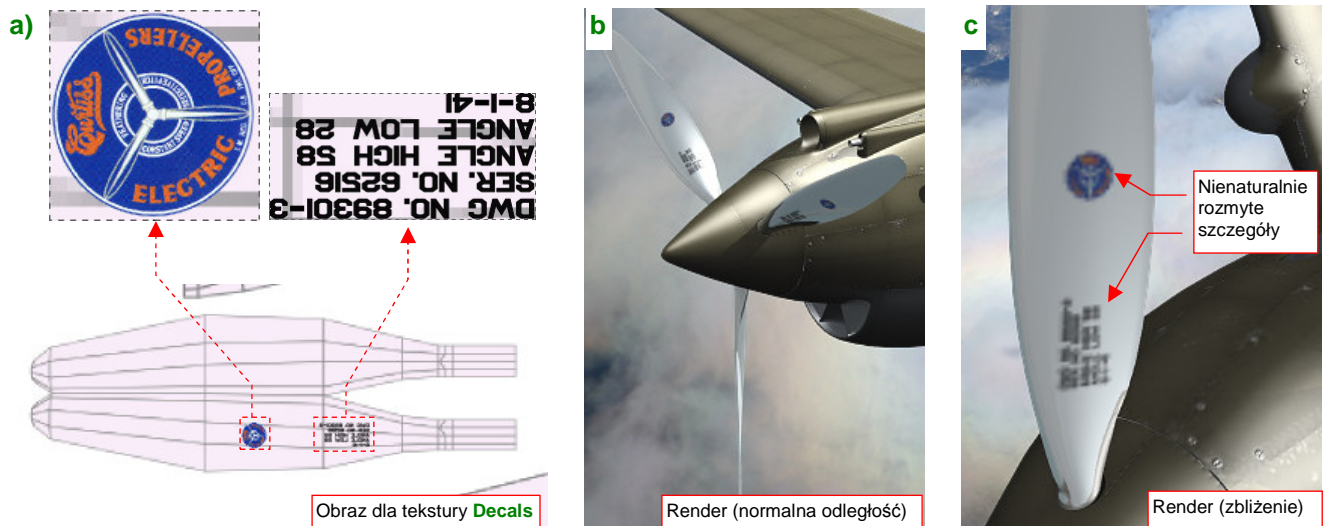
- Figurę pilota masz przygotowaną w odrębnym pliku Blendera (by można ją było wykorzystać w wielu modelach). Jest przypisana do warstwy 8. Zaimportuj ją do modelu p-40 (str. 486);
- Model pilota trzeba dopasować do kabiny. Czasami oznacza to poprawki w szczegółach jej wyposażenia (str. 487). Czasami jednak trzeba zmienić „pozę” pilota (str. 488). Do zmiany pozycji figurki służy umieszczona wewnątrz armatura („kości”);
- Po dopasowaniu pilota do kabiny, trzeba poprawić parę szczegółów, takich jak uprząż spadochronu (str. 489), czy kabel radiowy hełmofonu;
- Ostatnim elementem, jakim pozostał, jest druga wersja pasów fotela pilota (do scen „w locie”). Skopiuj w tym celu na warstwę 8 wszystkie sprzączki i końcówki pasów, stworzone w poprzedniej sekcji (str. 481). Należy je rozmieścić w nowych miejscach na figurce pilota (str. 489). Potem wystarczy połączyć je ze sobą nowymi segmentami pasa. Te fragmenty są zazwyczaj dość proste, i wtedy nie trzeba wykorzystywać do ich formowania krzywych.



## 7.7 Pozostałe szczegóły zewnętrzne

Do wykończenia pozostało jeszcze trochę drobiazgów. Aby o żadnym nie zapomnieć, proponuję zacząć je uzupełniać od nosa samolotu. Zasada jest podobna do tej, którą stosowaliśmy w kabinie (por. str. 477): sukcesywnie posuwamy się w kierunku ogona, tworząc po drodze wszystkie brakujące detale. Każdy z nich należy wykończyć nie pomijając materiałów i tekstur, by już nie trzeba było do niego więcej wracać. To także dobra okazja, by po kolei sprawdzić wygląd wszystkich zakamarków naszego modelu.

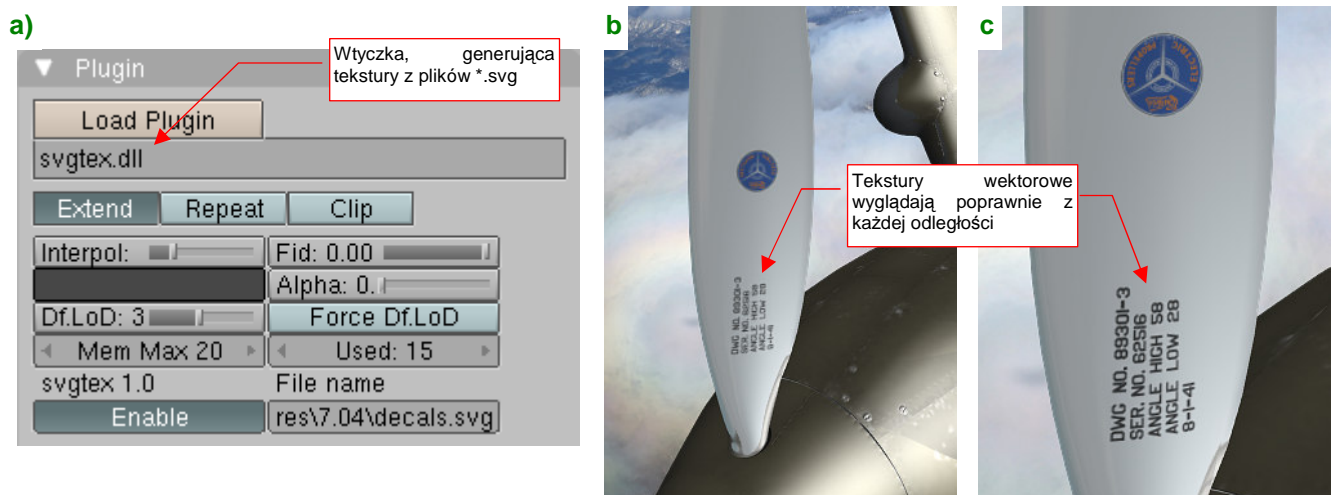
Zacznijmy od śmigła: na łopaty należy nałożyć teksturę z metryczką części i znakiem firmowym Curtiss. Jak pozostałe elementy tego typu, wkomponujemy ją w teksturę **B.Skin.Decals** (Rysunek 7.7.1a,b, por. str. 399):



Rysunek 7.7.1 Drobne oznaczenia (tekstura Decals) — problemy ze zbliżeniami

Okrągły emblemat „Curtiss Electric Propellers” możesz zeskanować z jakiegoś schematu malowania, pomniejszyć i wstawić do rysunku *decals.svg*. W tej dość „surowej”, rastrowej postaci powinien zadowalająco wyglądać na ujęciach z normalnej odległości (Rysunek 7.7.1b). Jeżeli chcesz, możesz poprawić jego jakość, poddając „wektoryzacji” (jak — patrz str. 746). Jednak mimo tych wysiłków, żadna rastrowa tekstura nie będzie dobrze wyglądać na zbliżeniach (Rysunek 7.7.1c). Akurat łopaty śmigła mogą być renderowane z takiej odległości: to ulubiony element fotografów, przy którym każą pozować pilotom... Czy coś z tym możemy zrobić?

Na szczęście — tak. Możesz zastosować specjalną wtyczkę Blendera, która umożliwia wykorzystanie wektorowych rysunków SVG jako tekstur. Tekstury, które generuje, „nie boją” się żadnych zbliżeń (Rysunek 7.7.2):



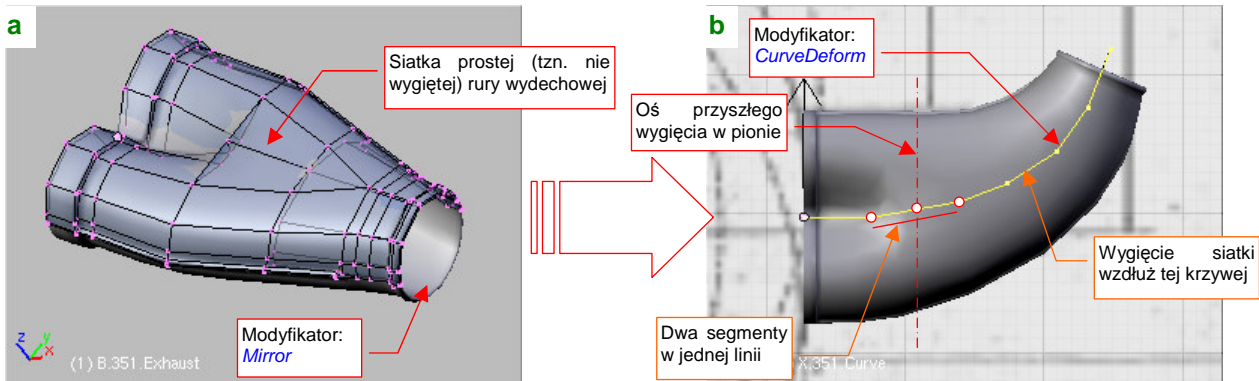
Rysunek 7.7.2 Rozwiązanie problemu zbliżeń za pomocą tekstury wektorowej

Więcej o użyciu tekstur wektorowych znajdziesz na str. 1017 i 613. Takie rozwiązanie ma także pewne wady.

Podstawową wadą tekstur wektorowych jest mniejsza „przeñośność” modelu. Oprócz pliku *\*.blend* trzeba także dostarczyć wtyczkę. Obecnie kod binarny tego komponentu działa wyłącznie w środowisku Windows. Oznacza to, że tego efektu nie można osiągnąć np. na Blenderze pracującym pod Linuxem.

Wróćmy do modelu i jego tekstur rastrowych. Na kołpaku śmigła namalowałem jeszcze trochę zacieków (na obrazach *color.png* i *ref.png*). Teraz można przejść do kolejnego obszaru: okapotowania silnika. Do tej pory brakowało na nim kolektora spalin, teraz wreszcie nadszedł czas, by go wykonać.

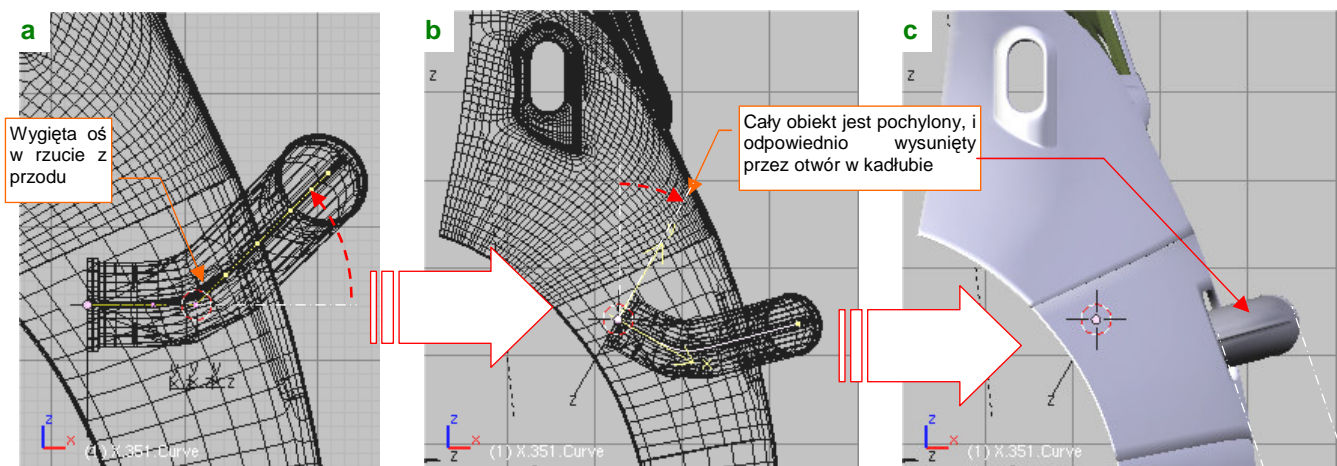
Kolektor zacznę od uformowania pojedynczej rury wydechowej. Takie rury miały dość „pogięty” kształt. Aby nie namęczyć się nad nim bez potrzeby, przygotowałem najpierw „wyprostowaną” wersję tego elementu (Rysunek 7.7.3a):



Rysunek 7.7.3 Formowanie pojedynczej rury wydechowej

Na takiej siatce zdecydowanie łatwiej jest kształtować odpowiednie przejścia pomiędzy kolejnymi sekcjami. Aby sprawę jeszcze bardziej uprościć, wykonuję tylko górną połowę siatki — dolną generuje mi modyfikator *Mirror* (Rysunek 7.7.3a). Potem wystarczy dodać modyfikator *Curve Deform*, związany z odpowiednią krzywą (*Path*), by wygiąć rurę w kształt, który widać na zdjęciach i rzutach z góry (Rysunek 7.7.3b).

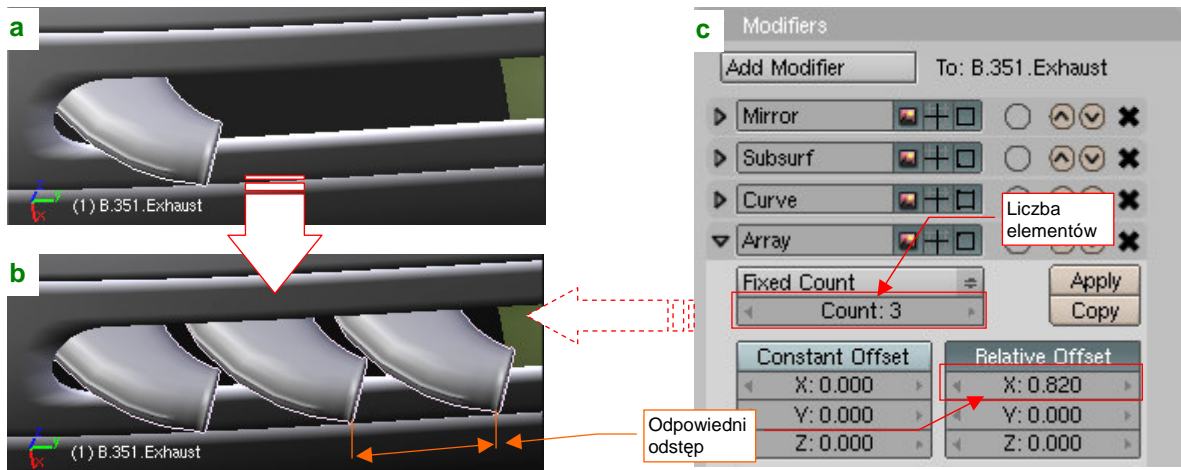
Na tym jednak formowanie się nie kończy, bo rura była wygięta także w płaszczyźnie prostopadłej (Rysunek 7.7.4a). To lokalne zagięcie, na przestrzeni dwóch segmentów wieloboku sterującego krzywą. Aby modyfikator *Curve Deform* nie skrzył nam siatki wokół tak „pogiętej” osi, w rzucie z góry ustawiłem te dwa odcinki w jednej linii (por. Rysunek 7.7.3b).



Rysunek 7.7.4 Kolektor spalin — ustalenie położenia w kadłubie

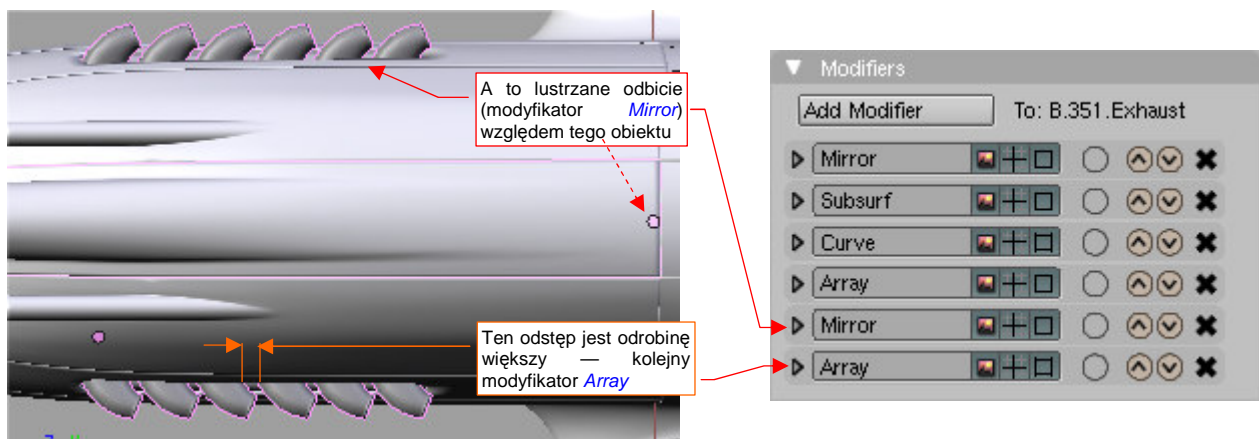
Gdy siatka jest już powyginana na wszystkie strony, pozostaje ją ustawić we właściwym miejscu. Zaczęłem od odpowiedniego pochylecia rury (tym razem jako obiektu, a nie siatki) (Rysunek 7.7.4b). Obróciłem ją o 30°, bo przypuszczałem że pod takim kątem była pochylona boczna ścianka cylindrów silnika, do której był przymocowany kolektor. Potem pozostało odpowiednio go ustawić wewnątrz kadłuba, by wystawał z niego na właściwą odległość. (Jaką? Co tu dużo gadać, na planach była za duża, dobrałem ją na podstawie zdjęć).

Mamy już pierwszą rurę, teraz należy ją powielić. Użyłem tu innego modyfikatora: **Array** (Rysunek 7.7.5a,b):



Rysunek 7.7.5 Powielenie rury wydechowej (modyfikator **Array**)

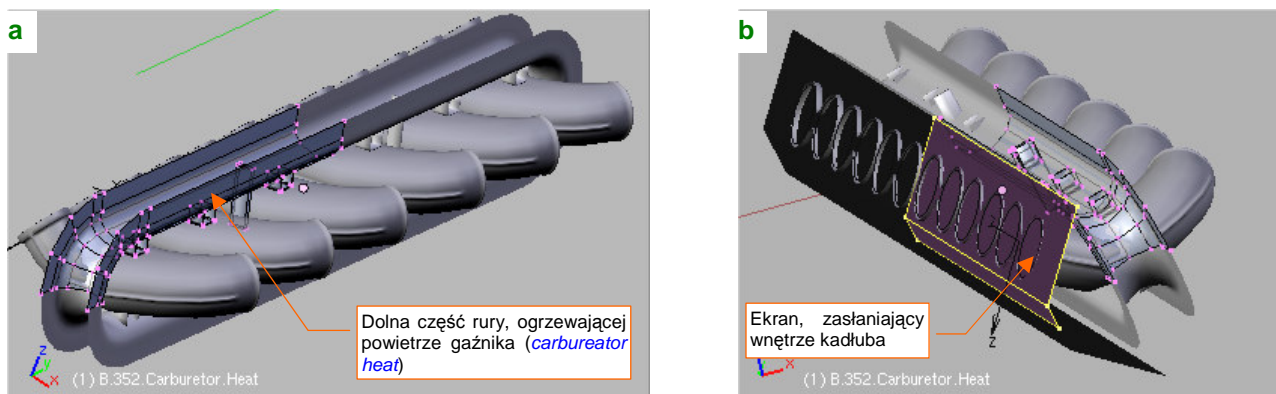
Wystarczyło go dodać i wpisać odpowiedni odstęp (**Offset** — por. Rysunek 7.7.5c), by pojawiły się kolejne rury! Drugi rząd rur (z prawej strony kadłuba) można wygenerować za pomocą kolejnego modyfikatora **Mirror** (Rysunek 7.7.6). Wystarczy podać jako punkt odniesienia środek innego obiektu (np. okapotowanie silnika):



Rysunek 7.7.6 Formowanie całości kolektora (wyłącznie za pomocą modyfikatorów!)

Dlaczego w poprzednim kroku wygenerowałem tylko trzy, a nie sześć rur? Bo pozostało coś dla koneserów: odstęp pomiędzy trzecim i czwartym cylindrem silnika Allison był nieco większy! Załatwi to kolejny modyfikator **Array** (Rysunek 7.7.6, a szczegóły o modyfikatorze **Array** — str. 923 i następna).

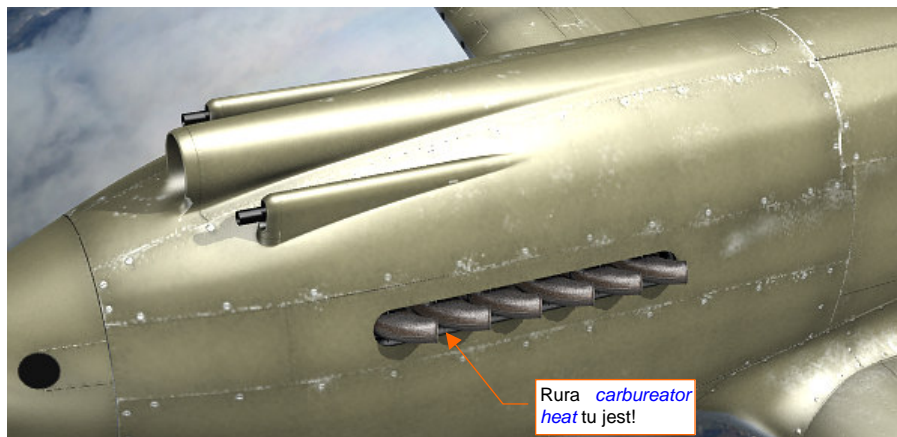
Pozostało już tylko „owinąć” wokół kolektora połowę rury ogrzewającej powietrze gaźnika (Rysunek 7.7.7a):



Rysunek 7.7.7 „Obudowa” kolektora spalin — ogrzewanie powietrza dla gaźnika

Na koniec dodałem z tyłu czarny „ekran”, by nie straszyć prześwitami w modelu (Rysunek 7.7.7b).

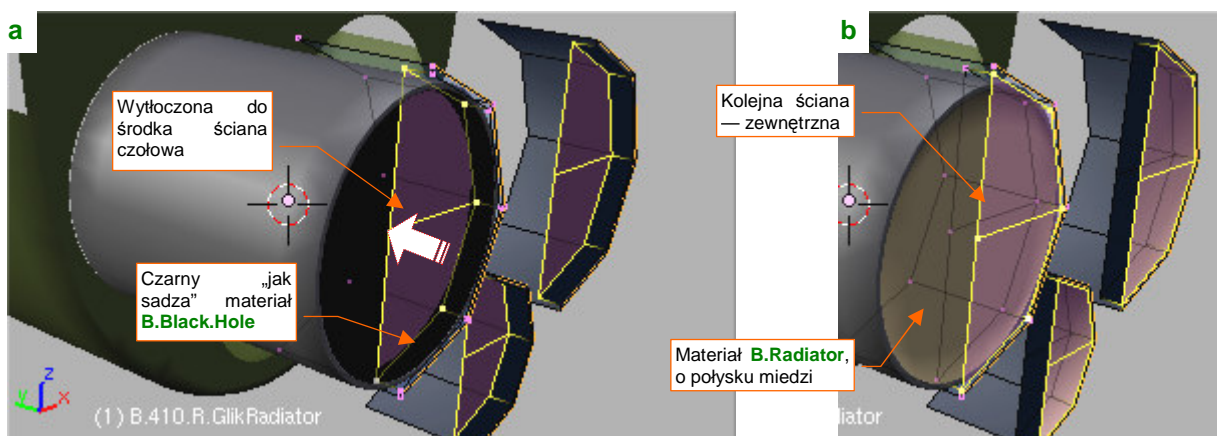
Kolektor pokryłem nowym materiałem **B.Steel.Exhaust**. To przerobiony **B.Steel.Rough**, w którym nałożyłem, za pomocą kolejnej tekstury proceduralnej „brudno-rdzawe” naleciałości (Rysunek 7.7.8):



Rysunek 7.7.8 Gotowy kolektor spalin — testowy render

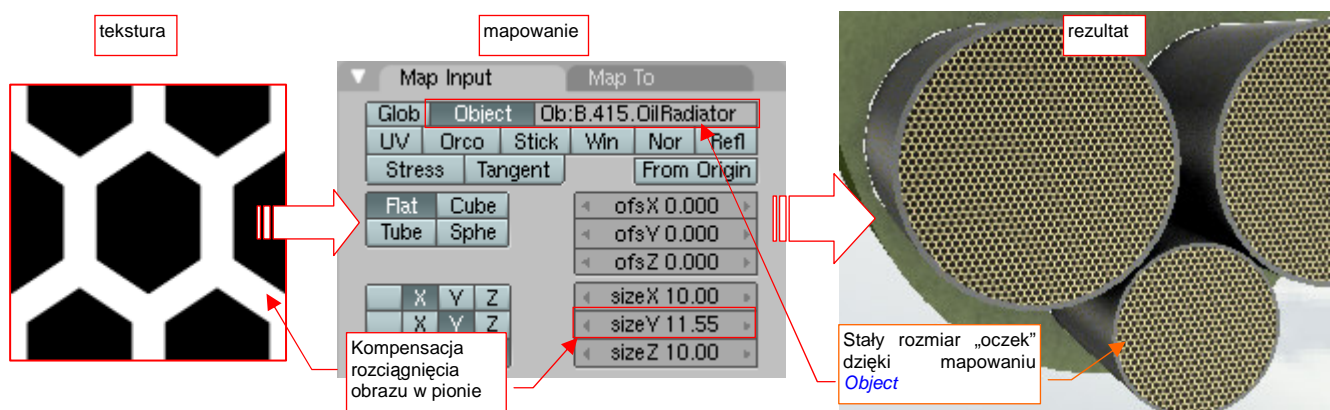
Rura z powietrzem dla gaźnika biegnie wewnątrz kadłuba, tuż przy krawędzi otworu. Nie jest na tej ilustracji specjalnie widoczna. Musisz mi uwierzyć na słowo, że pokryłem ją zwykłą stalą — materiałem **B.Steel.Rough**.

Kolejnym elementem, który wykończymy, jest zespół chłodnic. To ta sama siatka, powielona trzy razy. Ściankę czołową wytłoczyłem na niewielką głębokość do wnętrza, i pokryłem czarnym jak sadza, i podobnie wchłaniającym światło materiałem **B.Black.Hole** (Rysunek 7.7.9):



Rysunek 7.7.9 Poprawianie siatki chłodnic

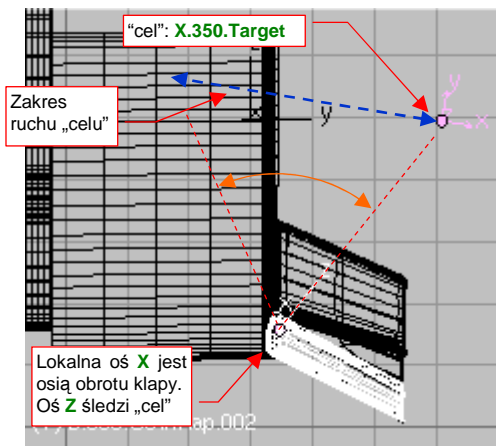
Następnie zasłoniłem tę wnękę ścianą zewnętrzną, którą pokryłem „zażółconą” odmianą **B.Steel.Smooth** (ma wyglądać jak mosiądz lub miedź). Przede wszystkim jednak ta ściana ma być perforowana, więc dodałem do tego materiału wzór „plastra miodu” w charakterze tekstury przejrzystości (*Alpha*) (Rysunek 7.7.10):



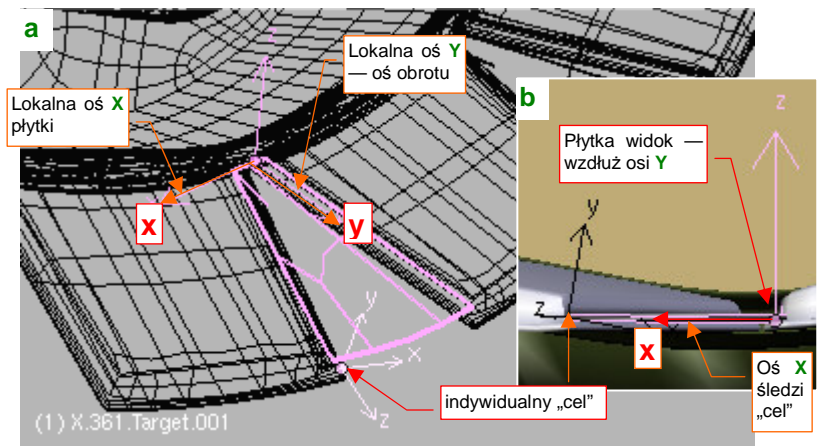
Rysunek 7.7.10 Odzworowanie perforacji na wlocie (i wylocie) z chłodnicy (materiał **B.Radiator**)

Materiał, imitujący wygląd ścianek chłodnicy (por. Rysunek 7.7.10), nosi nazwę **B.Radiator**. Otrzymał „miedziany” połysk, gdyż chłodnice Prestone, stosowane w P-40 aż do wersji „M”, były miedziane lub mosiężne<sup>1</sup>. Miały ścianki ułożone w sześciokątny wzór „plastra miodu”, który odtworzyliśmy za pomocą tekstury **B.Honeycomb**<sup>2</sup>. Rozmiar tych ścianek był taki sam, dla każdego rozmiaru chłodnicy. Aby taki efekt uzyskać, zamapowałem teksturę **B.Honeycomb** w trybie *Object*, dopasowując jej ostateczny rozmiar do najmniejszej chłodnicy oleju. (To ta w środku, o nazwie **B.415.OilRadiator**).

Skoro chłodnice są już gotowe, czas popracować nad ich tunelem wylotowym. Na jego obwodzie umieszczono cztery kłapy. „Zmechanizowałem” je za pomocą ograniczenia *Locked Track*, śledzącego pojedynczy obiekt - cel **X.360.Target** (Rysunek 7.7.11):



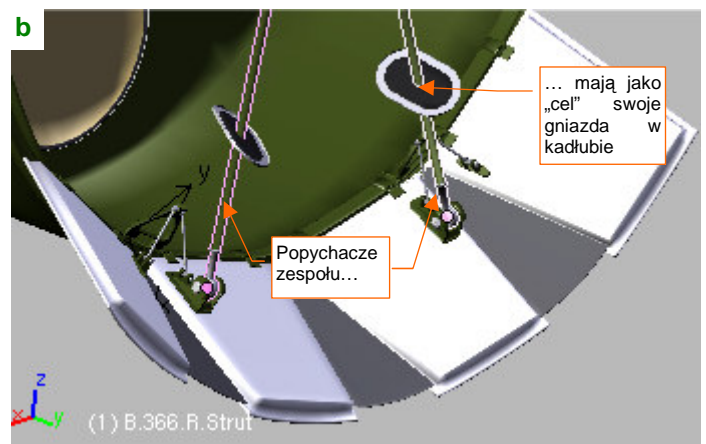
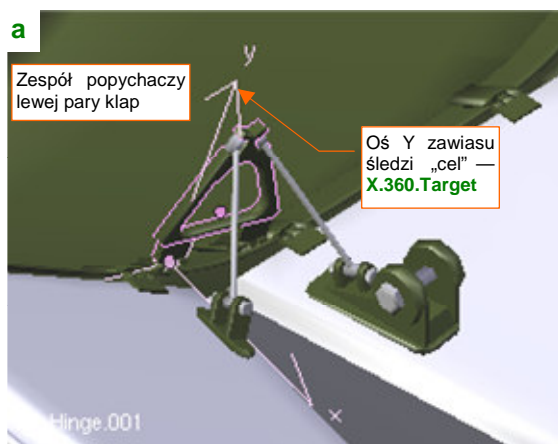
Rysunek 7.7.11 „Mechaniczacja” kłap chłodnic



Rysunek 7.7.12 Dopasowanie „uszczelnień” pomiędzy kłapami

Aby pojawiające się w szczelinach trójkątne „zastłonki” nie rozjeżdżały się podczas ruchu kłap, musiałem także dodać im ich własne ograniczenia *Locked Track* (Rysunek 7.7.12a). Ich osie **X** śledzą indywidualne „cele”, umieszczone na krawędzi sąsiedniej kłapy (Rysunek 7.7.12b).

Podobnie jak w przypadku podwozia, ruch obiektu-celu (**X.360.Target**) wymusza, za pomocą kolejnych ograniczeń *Locked Track*, ruch popychaczy. Zamodelowałem dwie pary takich elementów, by udawały, że to one poruszają kłapami (Rysunek 7.7.13):

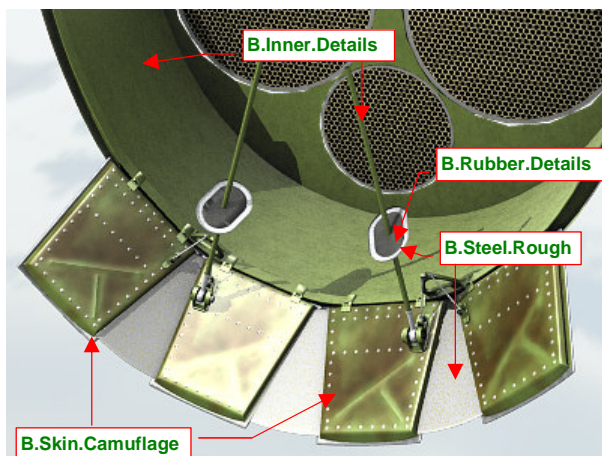


Rysunek 7.7.13 Odwzorowanie mechanizmu sterującego kłapami

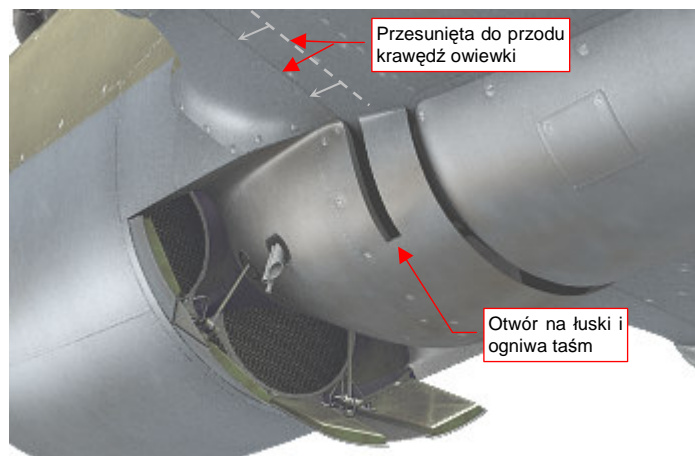
<sup>1</sup> W późniejszych konstrukcjach, jak choćby P-51 „Mustang”, czy nawet P-40N, napotkasz lżejsze chłodnice aluminiowe. Te chłodnice miały wzór ścianek zbliżony do tego, który do dziś jest używany w silnikach samochodowych, chłodzonych cieczą. Tak więc, gdy będziesz robił ich modele, w poszukiwaniu referencji wystarczy zajrzeć pod maskę własnego auta!

<sup>2</sup> Plik jest tak prosty i mały — 128x128 pikseli — że „spakowałem” go do pliku Blendera, a nie zostawiłem na dysku, jak inne obrazy. Wątpliwe, by trzeba było go w przyszłości zmieniać. W razie czego — źródłowy, wektorowy plik [honeycomb.svg](#) znajdziesz w materiałach źródłowych, folderze [source/p40/textures/7.07/src](#).

Jeżeli chodzi o dobór materiałów, to ostatecznie wnętrze kanału wylotowego pokryłem standardowym **B.Inner.Details** (Rysunek 7.7.14)<sup>1</sup>. Można tak zrobić, gdyż te powierzchnie są mało widoczne. Tym samym materiałem pokryte są części mechanizmu poruszającego całym zespołem — bo są małe. Otwory w kadłubie były zakryte gumowymi uszczelkami (**B.Rubber.Details**), z metalowym obramowaniem (**B.Steel.Rough**) — przynajmniej na rekonstrukcji z Duxford! W kolorze stali pozostawiłem także trójkąty zamykające szczeliny pomiędzy klapami. Za to same klapki, jako elementy najlepiej widoczne, pokryte zostały materiałem **B.Skin.Camouflage**, z odpowiednimi teksturami nierówności, odblasku i barwy. Pobrudziłem je na brązowo, zakładając jakieś drobne wycieki oleju z instalacji chłodnicy. Ostateczny rezultat przedstawia Rysunek 7.7.15:



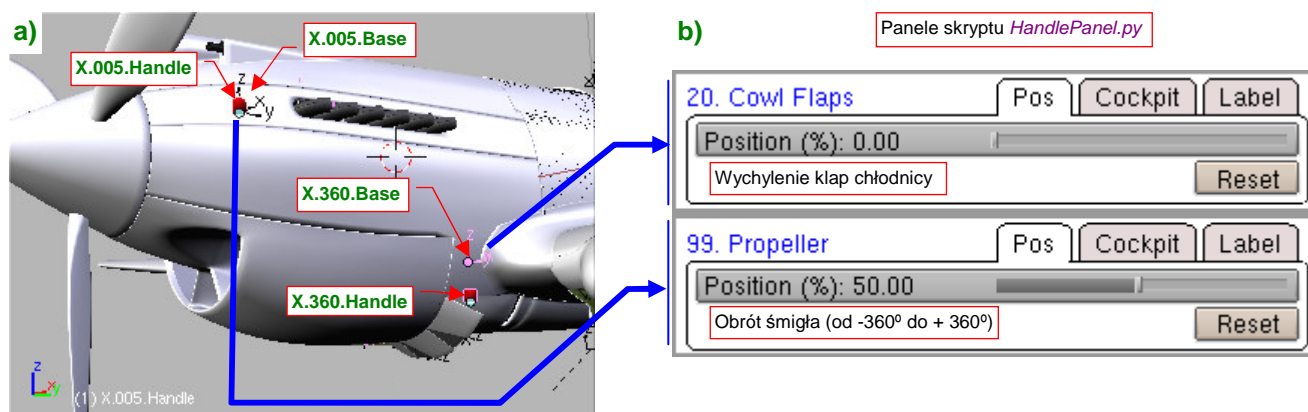
Rysunek 7.7.14 Wylot z chłodnicy — dobór materiałów



Rysunek 7.7.15 Ukończony wylot z chłodnicy (testowy render)

Zwróć uwagę na otwory na łuski, umieszczone za chłodnicą (Rysunek 7.7.15). Dodałem ten szczegół po zebraniu i przeanalizowaniu większej liczby zdjęć. Przeróbka siatki modelu na tym etapie była kłopotliwa, szczególnie związane z nią przesunięcie do przodu krawędzi owiewek skrzydeł, razem z teksturą śrub!

Zdecydowałem się dodać do paneli sterowania modelem (*HandlePanel.py*) dwa zestawy kontrolki, związane z nosem samolotu: jedną dla otwierania i zamykania klap, a drugą — dla obrotu śmigła (Rysunek 7.7.16b) :

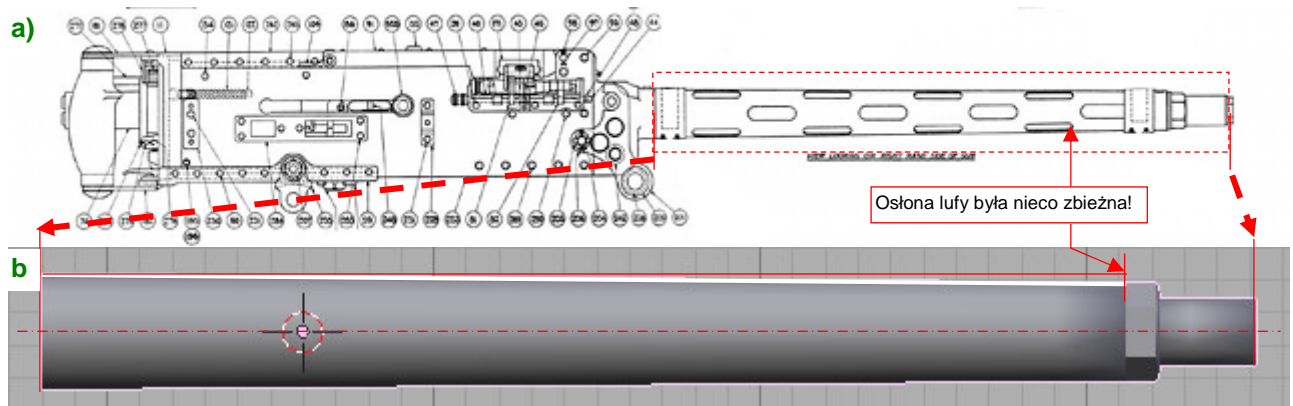


Rysunek 7.7.16 Nowe kontrolki, sterujące obrotem śmigła i klapami chłodnicy

Kontrolki tworzę tak samo, jak to robiliśmy dla podwozia (por str. 435 i dalsze). Najpierw wstawiam na warstwę 9 „punkt odniesienia”, przymocowany na trwałe (relacją *Parent*) do kadłuba. Dla obrotu śmigła to pusty obiekt (*Empty*) **X.005.Base**, a dla klap — **X.360.Base** (Rysunek 7.7.16a). Na warstwę 10 dodałem uchwyty **X.005.Handle** i **X.350.Handle**. Są powiązane z **\*.Base** relacjami *Parent* i ograniczeniami *Limit Location*. Z tymi uchwytami, za pomocą ograniczeń *Transfrom*, jest skojarzony ruch elementów modelu. Dla śmigła to po prostu obrót kołpaka (**B.005.Hub**), a dla klap — przesunięcie wzdłuż lokalnej osi **X** obiektu **B.360.Target** (por. str. 495).

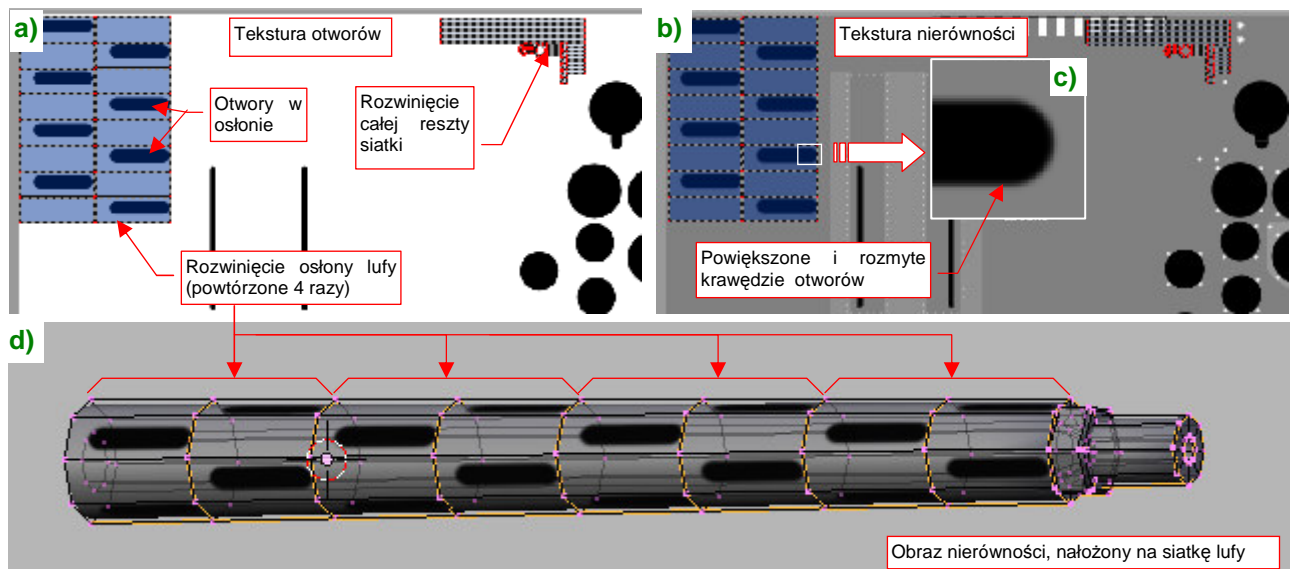
<sup>1</sup> Wcześniej był przewidziany dla **B.Skin.Camouflage** (por. str. 357). Dzięki temu zwolniło mi się miejsce na mapie **UV**. Wykorzystałem je na wstawienie większego rozwinięcia tarczy koła. (W oryginalnej, „małej” wersji, śruby na jej powierzchni wyglądały brzydko)

Przygotowując się do odwzorowania luf karabinów, zamontowanych w skrzydłach, ściągnąłem ze strony poświęconej karabinom Browing rysunek wersji lotniczej, kalibru 0.303 cala (Rysunek 7.7.17a) :



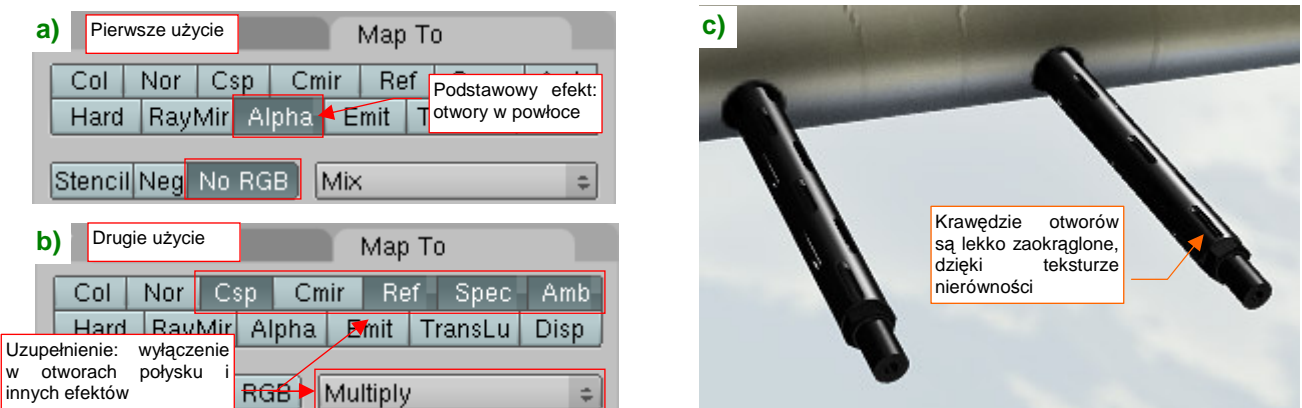
Rysunek 7.7.17 Odwzorowanie kształtu lufy karabinu Browing 0.303 cala

Z zaskoczeniem stwierdziłem, że zewnętrzna osłona lufy była lekko stożkowa! Oczywiście zaraz to odwzorowałem (Rysunek 7.7.17b). Widoczne na rysunku otwory w osłonie wykonałem za pomocą dwóch tekstur: otworów (**B.Instruments.Holes**, Rysunek 7.7.18a) i nierówności (**B.Instruments.Nor**, Rysunek 7.7.18b):



Rysunek 7.7.18 Przygotowanie otworów w osłonie lufy

Lufy wykorzystują ten sam materiał co zamki w kabinie pilota: **B.Instruments.Transp**, bo jest „odpowiednio czarny”. W tym materiale zwróć uwagę na dwukrotne użycie tej samej tekstury otworów (Rysunek 7.7.19a,b):



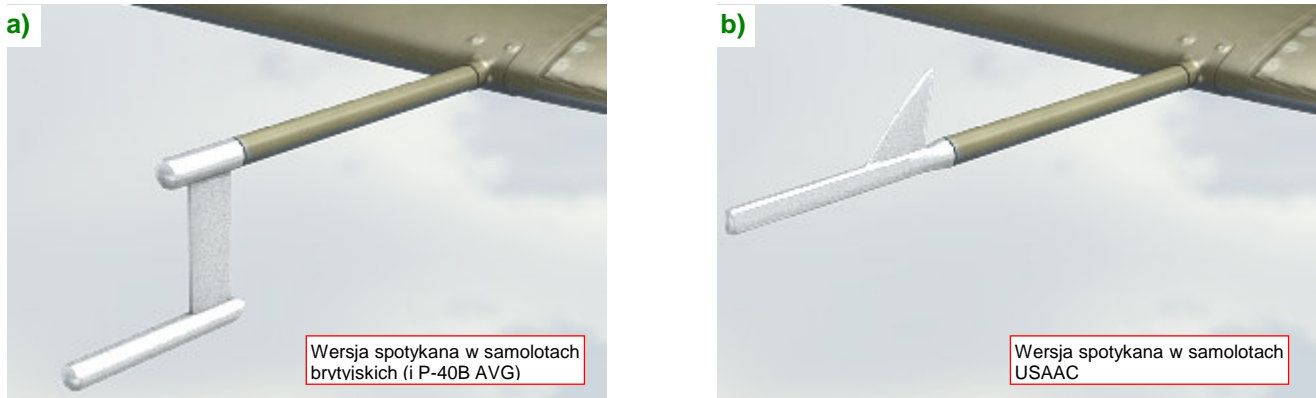
Rysunek 7.7.19 Szczegóły dwukrotnego mapowania tekstury otworów **B.Instrument.Holes** w materiale **B.Instruments.Transp**.

Rysunek 7.7.19c pokazuje uzyskany rezultat — myślę, że jest wystarczający dla ujęć z większej odległości.

- Przy modelowaniu konkretnej maszyny zwróć uwagę, czy końce luf karabinów nie były zaklejone jakąś jaskrawą taśmą klejącą. Tak często zabezpieczano je przed kurzem, wzbijanym przez samolot podczas startu i lądowania. To już indywidualny szczegół samolotu, który w razie czego musisz uzupełnić.

Właśnie z tego powodu nie cyzelowałem specjalnie końcówek luf karabinów — i tak były zazwyczaj zasłonięte...

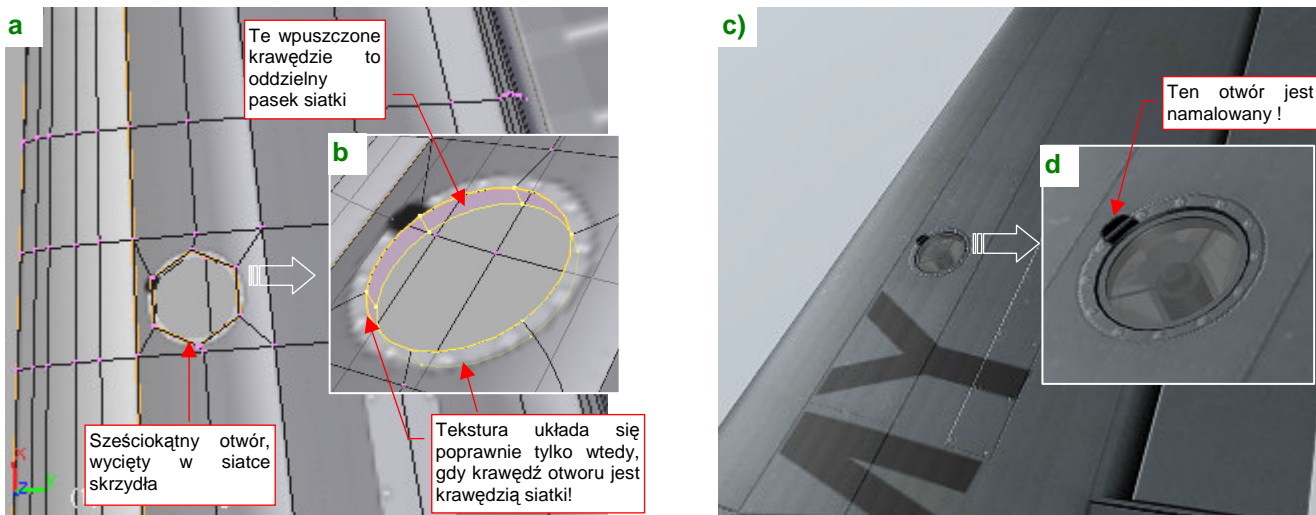
Kolejną „rurką” sterzącą z krawędzi natarcia samolotu jest czujnik prędkościomierza, nazywany od nazwiska wynalazcy „rurką Pitota”. Egzemplarze P-40 produkowane dla Wielkiej Brytanii miały inną końcówkę tego przyrządu (Rysunek 7.7.20a), niż samoloty wytwarzane na zamówienie USAAC (Rysunek 7.7.20b):



Rysunek 7.7.20 Dwie wersje rurki Pitota

Podobnie jak w przypadku reflektora celownika (por. str. 476), przygotowałem obydwie wersje, i umieściłem je na razie na różnych warstwach. W następnym rozdziale podzielimy ten plik Blendera na różne sceny: jedną z Pearl Harbor, inną z AVG. Umieścimy wtedy odpowiednią wersję w odpowiedniej scenie.

Innym szczegółem jest wysuwany reflektor do lądowania w nocy. Wbudowano go w lewe skrzydło. Zdecydowałem się wykonać w siatce tego płata rzeczywisty otwór, w oparciu o sześciokąt (Rysunek 7.7.21a). Ściany w tym miejscu są płaskie, więc na brzegach nie było niebezpieczeństwa deformacji:



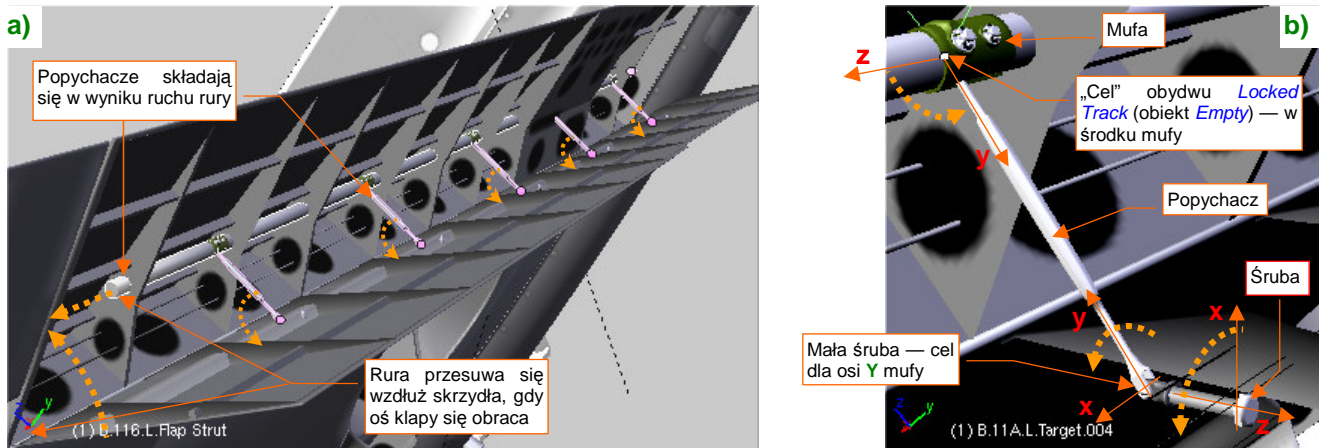
Rysunek 7.7.21 Składany reflektor do lądowania (w lewym płacie)

Po tej lokalnej zmianie siatki musiałem ponownie ułożyć w przestrzeni UV zmienione ściany by koło, narysowane na teksturze wokół otworu, wyglądało poprawnie. Gdy je uformowałem, zaznażyłem krawędź otworu jako „ostrą” (*Cearse* = 1) i wytłoczyłem wgłęb dodatkowy „pasek” (Rysunek 7.7.21b). Ku mojemu zaskoczeniu, obecność tych nowych ścian znów „powyginała” narysowany na teksturze okrąg! Żadne zaznaczanie szwów wzdłuż krawędzi nie pomogło, i jedynym rozwiązaniem okazało się fizyczne oddzielenie siatki paska od reszty skrzydła. Uformowanie samego reflektora nie było jakąś specjalną filozofią: ot, stożek z jakimś cylindrem (pełniącym rolę „żarówki”) i szybka. Testowy render reflektora i skrzydła przedstawia Rysunek 7.7.21c,d).



Kolejnym elementem płata jest mechanizm klap. W P-40 zastosowano kłapy „krokodylowe”. Po otwarciu takich klap wewnątrz skrzydła staje się widoczne, a w nim — mechanizm, który nimi porusza, oraz jakieś elementy szkieletu płata. Na szczęście już na etapie modelowania odwaliliśmy połowę roboty, umieszczając gdzie trzeba wszystkie żebra i podłużnice (por. str. 257). Teraz pozostał do dorobienia sam mechanizm.

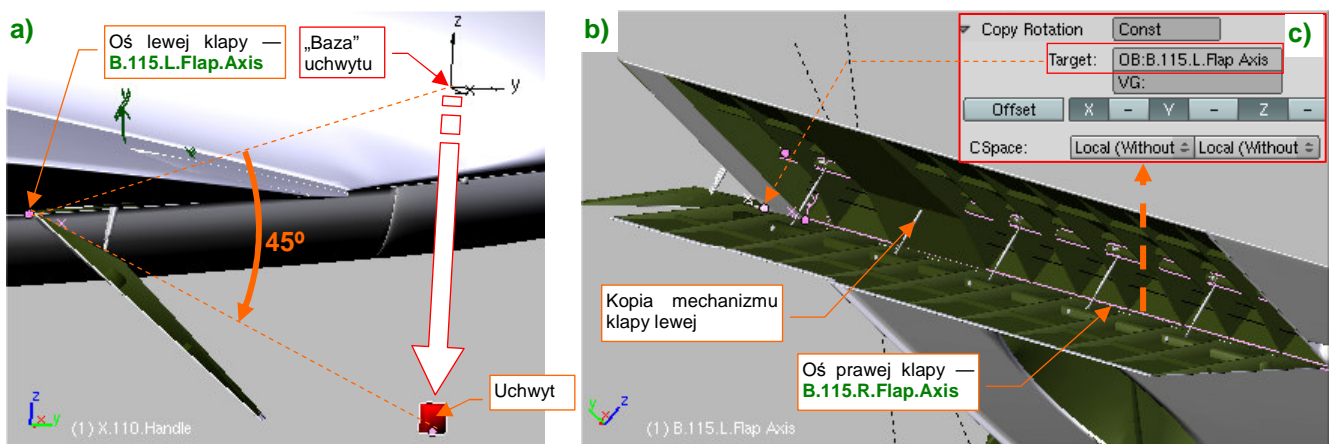
Mechanizm klap krokodylowych działał w każdej konstrukcji w ten sam sposób: wzdłuż skrzydła biegła rura, do której były przymocowane popychacze (Rysunek 7.7.22a). Drugi koniec każdego popychacza były przymocowane przegubowo do szkieletu kłapy (Rysunek 7.7.22b):



Rysunek 7.7.22 Mechanizm składający kłapę

Aby złożyć kłapę, wystarczyłoby siłownik hydrauliczny przesunął rurę wzdłuż jej osi podłużnej. Przesuwająca rura „ciągnęła” za sobą popychacze, a te — kłapę. Przy otwieraniu wszystko poruszało się po prostu w przeciwną stronę. Mechanizując nasz model jak zwykle zamieniłem przyczynę na skutek: tu kłapę porusza obracający się wałek zawiasów. Z obrotem walca skojarzone jest (ograniczeniem *Transformation*) odpowiednie przesunięcie rury (Rysunek 7.7.22a). Rysunek 7.7.22b) przedstawia zespół pojedynczego popychacza (aby go dobrze pokazać, kłapa jest na nim zupełnie przezroczysta). Na rurze umieściłem mufę. W środku mufy jest niewidzialny obiekt (*Empty*) — cel dla ograniczeń *Locked Track* popychacza i mocującej go śruby. Śledzi go, obracając się wokół *Z*, oś *X* śruby. Śledzi go także oś *Y* popychacza, obracając się wokół lokalnej osi *X*. Cała rura się tylko przesuwana, ale sama mufa się obraca, śledząc osią *Y* małą śrubę na drugim końcu popychacza.

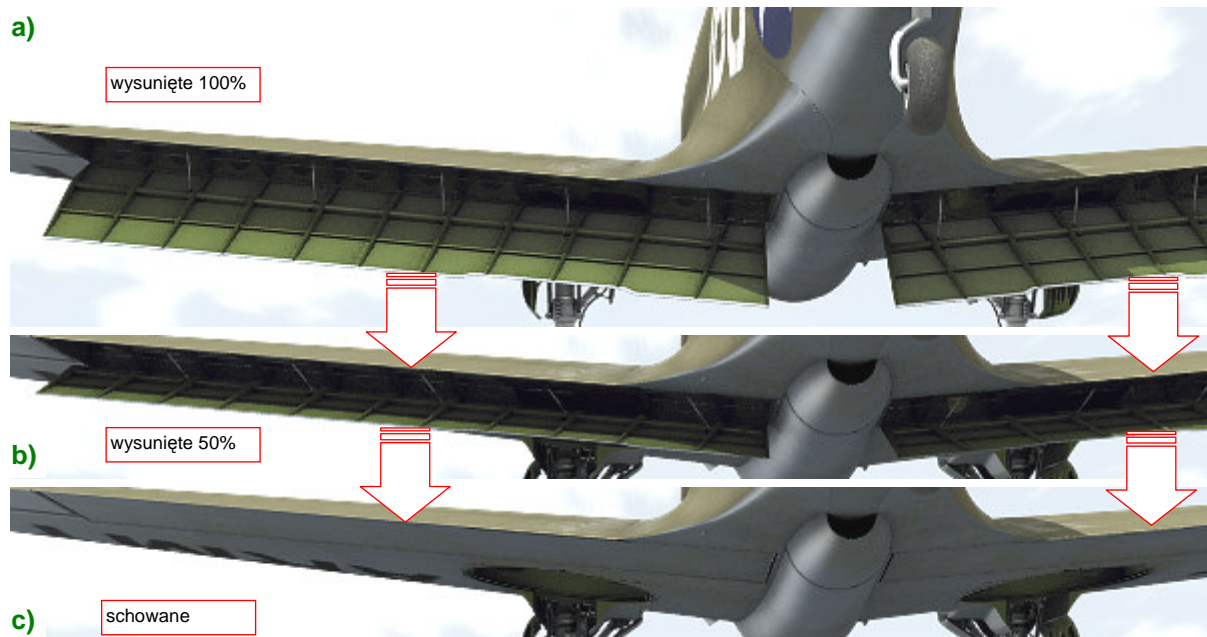
By ułatwić obrót osią kłapy, dodałem do modelu kolejny uchwyt (wraz z bazą — „rodzicem”) (Rysunek 7.7.23a):



Rysunek 7.7.23 Sterowanie kłapą za pomocą uchwyty

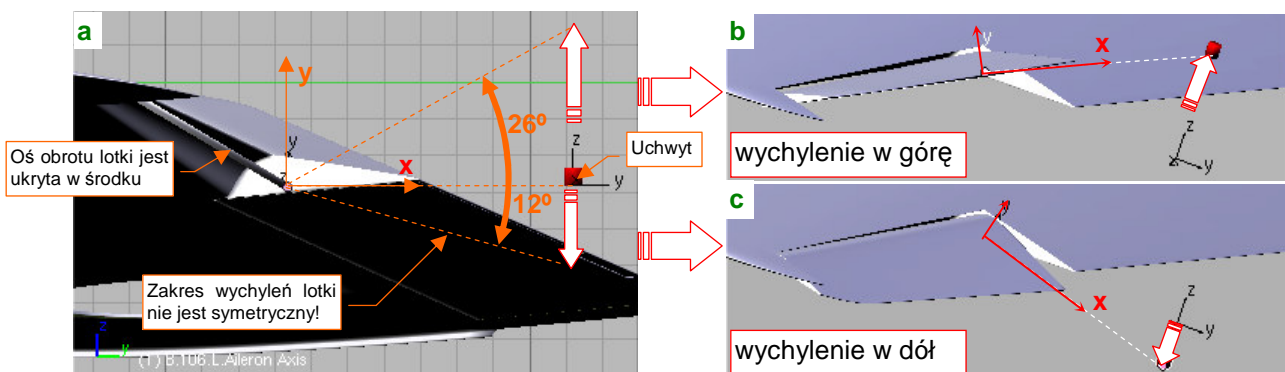
Szczegółowy opis ustalania krańcowych położenia dla takiego uchwytu znajdziesz na str. 824 i dalszych. Gdy mechanizm lewej kłapy zaczął działać, skopiowałem go do kłapy prawej. Okazało się, że z tym płatem jest trudniej, gdyż ma ujemną skalę (*ScaleX=-1*) i ograniczenie *Locked Track* działa tu zupełnie niepoprawnie. Na szczęście można było powiązać obrót osi prawej kłapy z lewą ograniczeniem *Copy Rotation* (Rysunek 7.7.23c).

Rysunek 7.7.24 przedstawia kolejne fazy chowania kłapy P-40. Zwróć uwagę na charakterystyczne „składanie” popychaczy na boki:



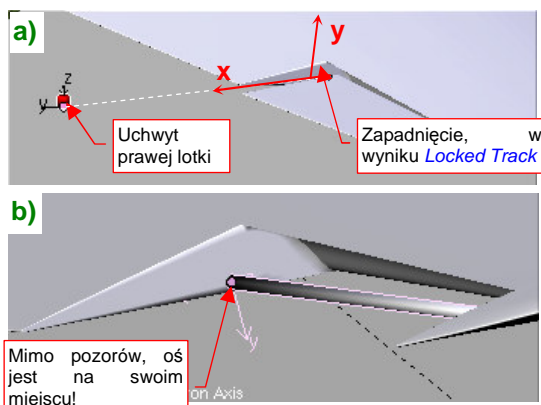
Rysunek 7.7.24 Kolejne fazy chowania kłapy

Lotki także mają swoje osi obrotu. Dodałem do lewej uchwyt, by nimi sterować (Rysunek 7.7.25):

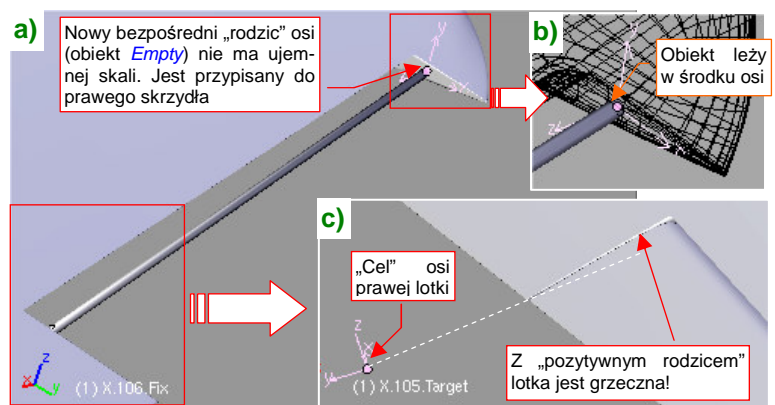


Rysunek 7.7.25 Sterowanie lewą lotką

Oś lotki śledzi za pomocą ograniczenia *Locked Track* aktualne położenie uchwytu. Lotki wychylają się niesymetrycznie (Rysunek 7.7.25b,c), więc uchwyt ma niesymetryczny zakres ruchu wzdłuż osi *Z* (Rysunek 7.7.25a). To rozwiązanie nie działa na prawym płacie: lotka z niewiadomych przyczyn „zapada się”! (Rysunek 7.7.26):



Rysunek 7.7.26 Problem z prawą lotką



Rysunek 7.7.27 Eliminacja wpływu ujemnej skali skrzydła na lotkę

To wpływ ujemnej skali prawego skrzydła. Pomogło dodanie nowego „rodzica” osi lotki (Rysunek 7.7.27).

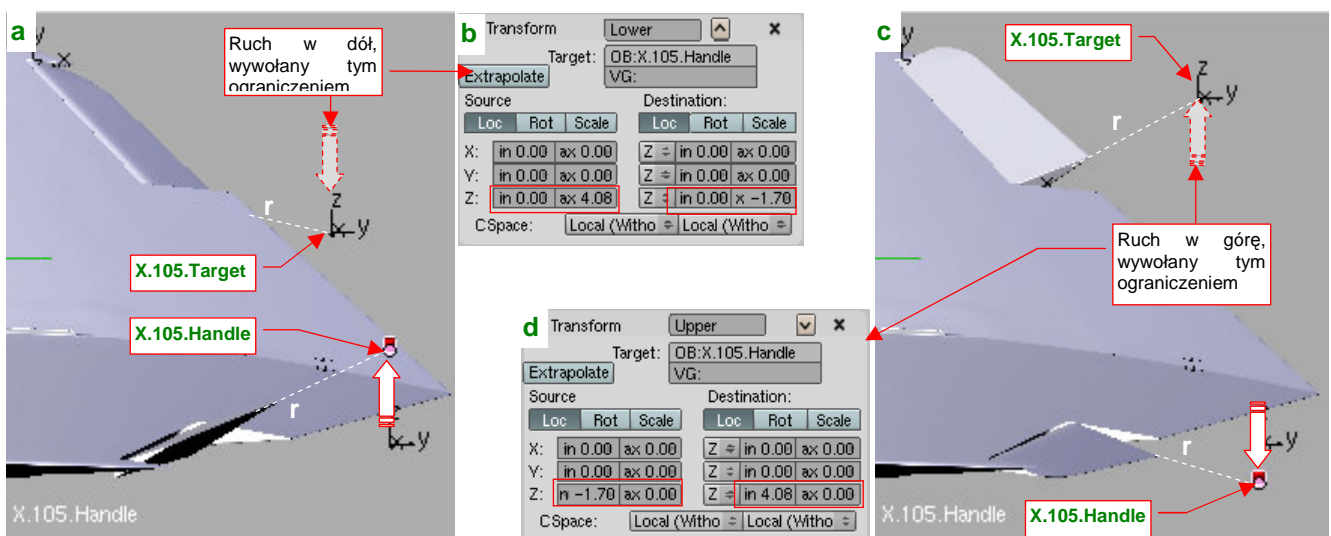
Rozwiązanie, które pokazuje Rysunek 7.7.27, polega na wstawieniu w strukturę hierarchii modelu (powiązań *parent-child*) nowego, bezpośredniego „rodzica” osi lotki. To pusty obiekt (*Empty*), przypisany relacją *parent* do prawego skrzydła. Zajmie w hierarchii miejsce, które poprzednio należało do osi. W odróżnieniu od płata, ten pomocniczy obiekt ma wszystkie skale dodatnie. Dzięki „pozytywnemu” rodzicowi, osi prawej lotki można przypisać poprawnie działające ograniczenie *Locked Track*.

- Postaraj się pamiętać, by przed rozwinięciem w UV siatki płatów wyeliminować obiekty o skali = -1. Można to zrobić przeskalowując ich siatki o -1 wzdłuż odpowiedniej osi, by cały obiekt nie miał ujemnej skali.

Jeżeli jednak o tym zapomniałeś — tak jak ja — musisz, radzić sobie z różnymi ubocznymi skutkami ograniczeń, nałożonych na takie „obiekty ujemne” jak prawe skrzydło. Nie ukrywam, że próbowałem na obecnym etapie prac zmienić jego skalę na dodatnią. Spowodowało to jednak „rozkraczenie” prawej nogi podwozia (tam też są użyte ograniczenia!). Ostatecznie wolałem się skoncentrować na skorygowaniu samej lotki, bo to było prostsze niż poprawki w mechanizmie składania koła głównego.

Dlaczego nie poradziliśmy sobie z tą lotką tak, jak to zrobiliśmy z klapą — stosując zamiast *Locked Track* ograniczenie *Copy Rotation* (por. str. 499)? Ponieważ lotki wychylają się różnicowo: gdy jedna porusza się w górę, to druga — w dół. W dodatku różne są kąty tych wychyleń: gdy prawa jest podniesiona do góry o  $26^\circ$ , to lewa jest obrócona w dół o  $12^\circ$ , i na odwrót (por. Rysunek 7.7.25)<sup>1</sup>.

Aby dokładnie odwzorować różnicowe wychylenia lotek, skierowałem (za pomocą ograniczenia *Locked Track*) oś prawej lotki na pomocniczy obiekt **X.105.Target** (Rysunek 7.7.28a):

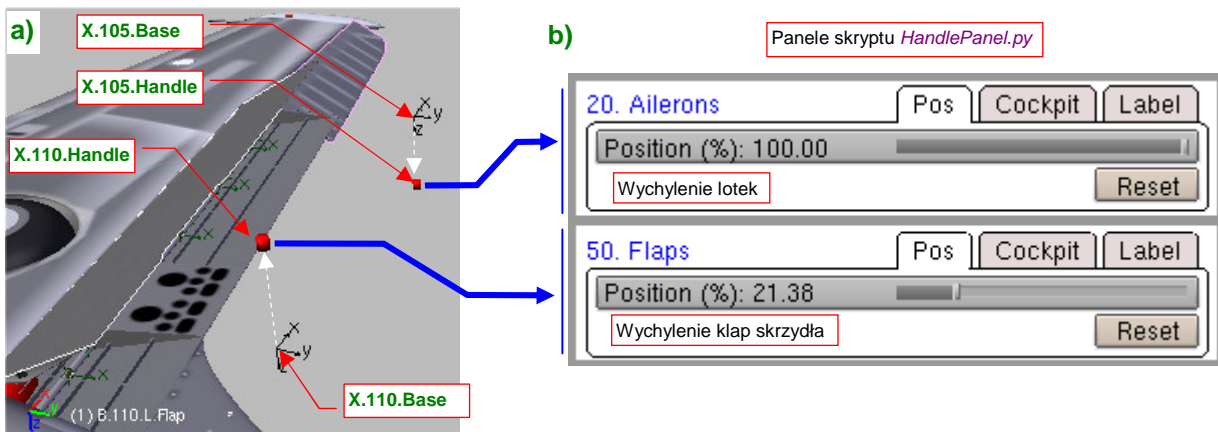


Rysunek 7.7.28 Różnicowe wychylenia lotek, uzyskane za pomocą dwóch ograniczeń *Transform*.

Gdy lotki nie są wychylone, „cel” **X.105.Target** znajduje się dokładnie w tym samym miejscu co uchwyt lewej lotki (**X.105.Handle**), ale po przeciwnej stronie samolotu (ich położenie różni się tylko znakiem współrzędnej **X**). Całym zespołem lotek steruje ruch uchwytu **X.105.Handle**. Położenie „celu” prawej lotki — **X.105.Target** — jest z nim powiązane za pomocą dwóch ograniczeń *Transform*. Pierwsze z nich nazwałem **Lower**, bo powoduje, że gdy **X.105.Handle** przesuwa się do góry o 4.08 jednostki, **X.105.Target** wykonuje ruch do dołu o 1.7 jednostki. (Rysunek 7.7.28a,b). Drugie ograniczenie — **Upper** — odpowiada za alternatywny ruch **X.105.Target** do góry (Rysunek 7.7.28c,d). (Wartość 4.08 to  $\tan(26^\circ) \cdot r$ , a 1.7 to  $\tan(12^\circ) \cdot r$ , gdzie  $r$  to odległość „celu” do osi lotki).

<sup>1</sup> Lotki wychylają się o różne kąty w górę i w dół, aby zapewnić (choć w przybliżeniu) podobne siły aerodynamiczne. Wynika to z niesymetrycznego profilu płata (górną część jest bardziej wypukła niż dolna) oraz faktu, że skrzydło jest zazwyczaj ustawione pod pewnym kątem do toru lotu samolotu. (Ten kąt, nazywany „kątem zaklinowania”, podczas lotu z dużą prędkością jest bardzo mały, ale może dochodzić do  $13\text{--}15^\circ$  podczas lądowania!)

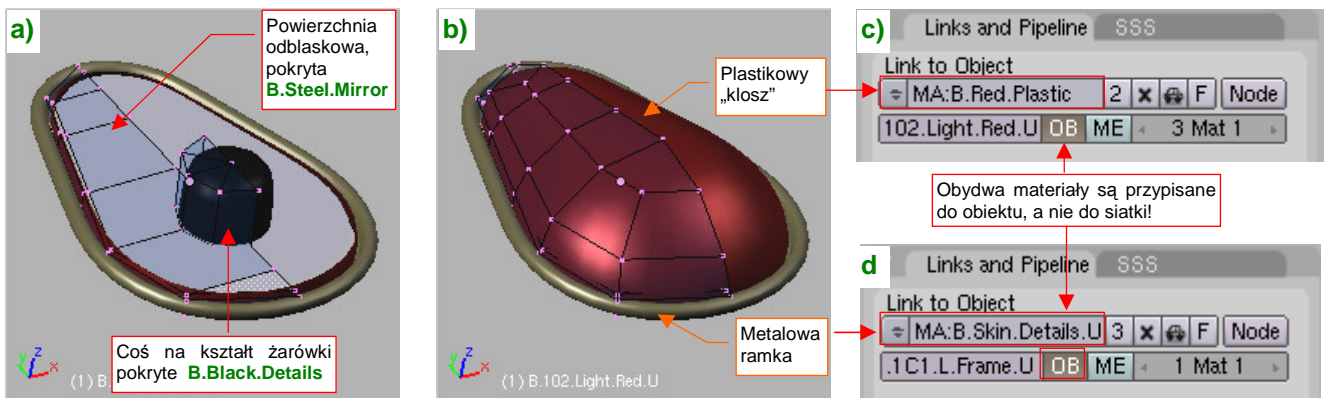
Na koniec pozostało nadać uchwytem lotek i klap odpowiednie oznaczenia na naszym panelu sterowania modelem (*HandlePanel.py*) (Rysunek 7.7.29):



Rysunek 7.7.29 Nowe kontrolki, sterujące klapami i lotkami

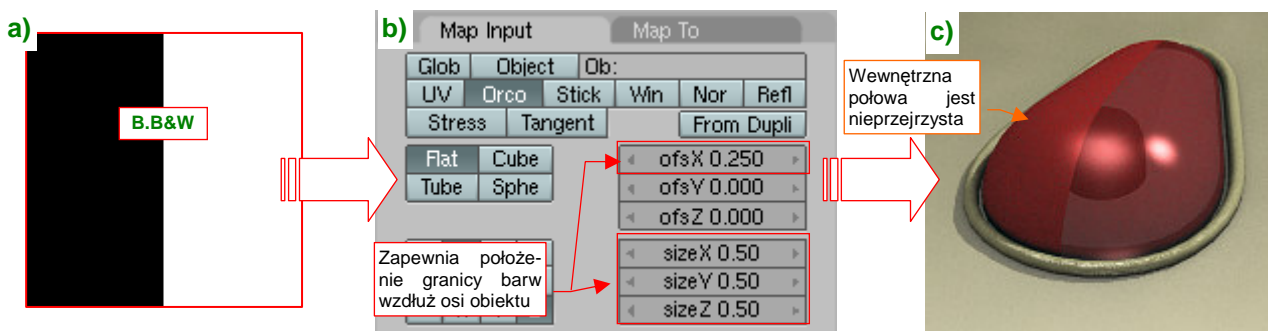
Obiekty „bazowe” (**X.105.Base**, **X.110.Base**) są „przyczepione” do lewego płata relacją *Parent*. Ich położenie wyznacza domyślne pozycje uchwytów **X.105.Handle** i **X.110.Handle** (Rysunek 7.7.29a). Przedrostki etykiet kontrolki („20.”, „50.” — por. Rysunek 7.7.29b) ustalam, starając się je ułożyć w logicznej kolejności. (Pomiędzy lotkami i klapami zostawiłem miejsce w numeracji na stery kierunku i wysokości).

Ostatnim szczegółem, który pozostał do ukończenia obszaru płatów, są światła pozycyjne. Element niewielki, ale wyeksponowany na końcówkach skrzydeł. Uformowałem najpierw ich podstawę (Rysunek 7.7.30a):



Rysunek 7.7.30 Światło sygnalizacyjne — szczegóły

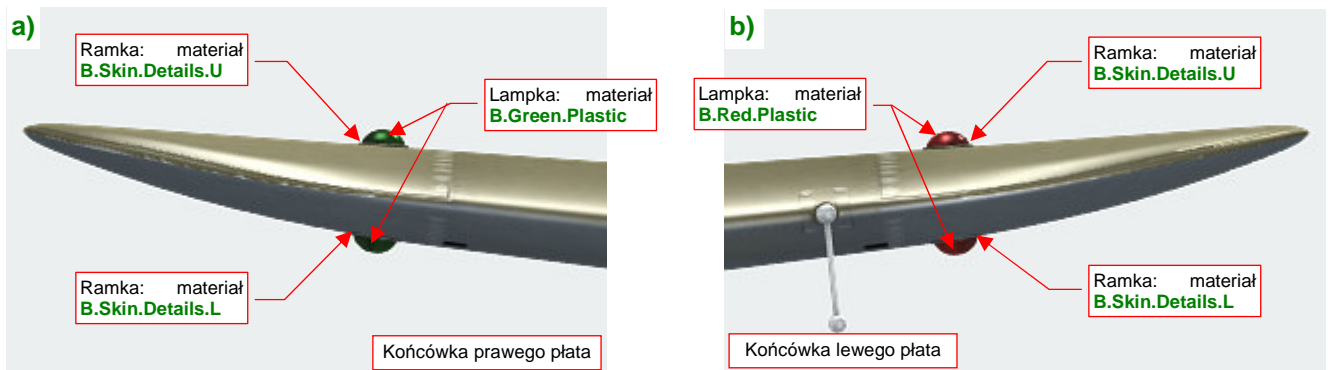
Potem zakryłem tę lampę „kloszem”, pokrytym materiałem **B.Red.Plastic** (Rysunek 7.7.30b). (Opis tego materiału znajdziesz na str. 983). P-40 miał sześć takich światel — po dwa z każdej strony skrzydeł i statecznika. Materiały plastiku i ramki pochodzą z przypisanego obiektu, a nie z siatki (Rysunek 7.7.30c,d) — by jej nie powielać. Osłony światel pozycyjnych były z jednej strony nieprzezroczyste, by nie oślepić pilota (Rysunek 7.7.31c):



Rysunek 7.7.31 Częściowe „zaślepienie” osłony lampki za pomocą tekstury (materiał **B.Red.Plastic**)

Uzyskałem ten efekt poprzez odpowiednie mapowanie (Rysunek 7.7.31b) tekstury **B.&W** (Rysunek 7.7.31a).

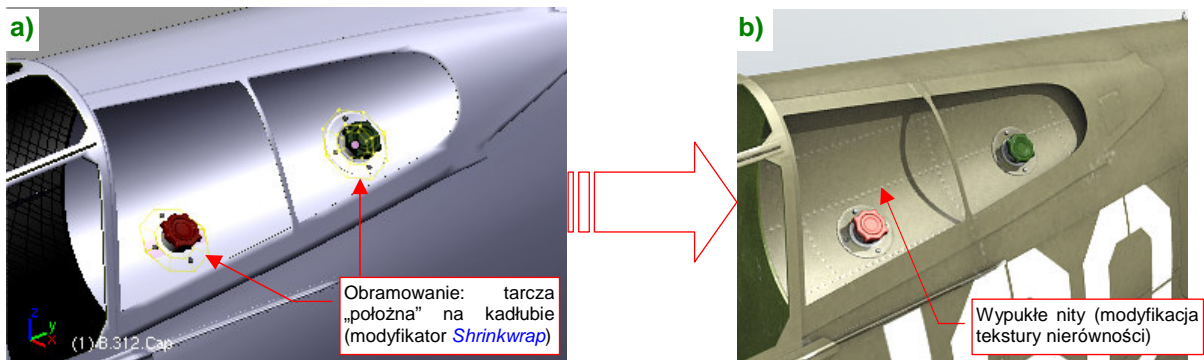
Inna sprawa, że cyzelowanie takiego szczegółu, jak lampka pozycyjna, wynika wyłącznie z czystej modelarskiej pasji. Na wszystkich renderach nie będzie tego widać — bo ten detal jest po prostu za mały (Rysunek 7.7.32):



Rysunek 7.7.32 Materiały, przypisane do poszczególnych obiektów (klonów) lampek pozycyjnych

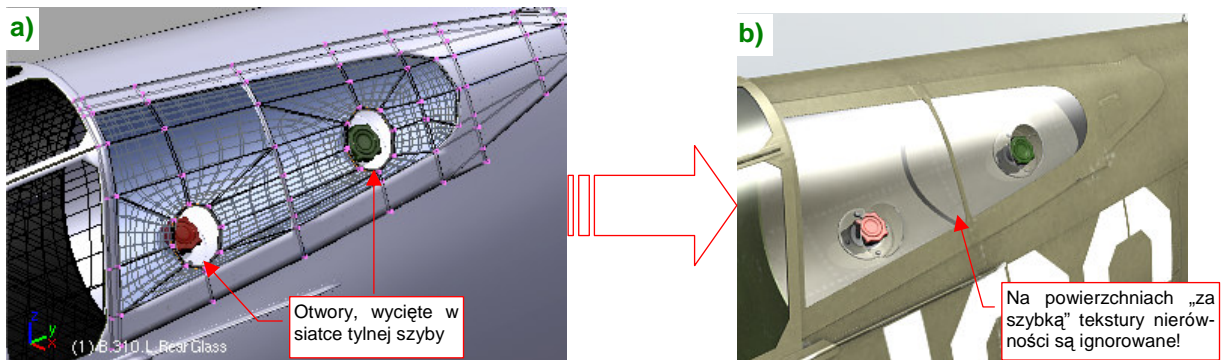
Rysunek 7.7.32a) pokazuje, jak wyglądają lampki na prawej końcówce płata, a Rysunek 7.7.32b) — na lewej. Prawda, że nie są duże? Zwróć za to uwagę, że mają różne barwy — mimo tego, że wszystkie cztery wykorzystują tę samą siatkę! (Dopóki jest to możliwe, staram się nie powielać siatek. To bardzo się przydaje, gdy za jakiś czas okaże się, że ich kształt trzeba poprawić). Możliwość zróżnicowania barw dla poszczególnych obiektów uzyskaliśmy w wyniku powiązania materiału „klosza” i ramki z obiektem, a nie siatką (por. str. 502, Rysunek 7.7.30). Ramek lamp nie rozwijałem w UV, tylko przypisałem im „uproszczone” wersje barw powierzchni wewnętrznych samolotu: szarą **B.Skin.Details.L** i oliwkowozieloną **B.Skin.Details.U**. (Podobne rozwiązanie zastosowaliśmy we wnętrzu kabiny). Tych dwóch barw użyjemy także do pokrycia innych detali zewnętrznych.

Wlew paliwa do zbiornika kadłubowego znajdował się w P-40 za kabiną pilota. W wersjach B/C były to dwa „korki”, umieszczone w ciągnących się za nią wyźłobieniach (Rysunek 7.7.33):



Rysunek 7.7.33 „Korki” tylnego zbiornika paliwa

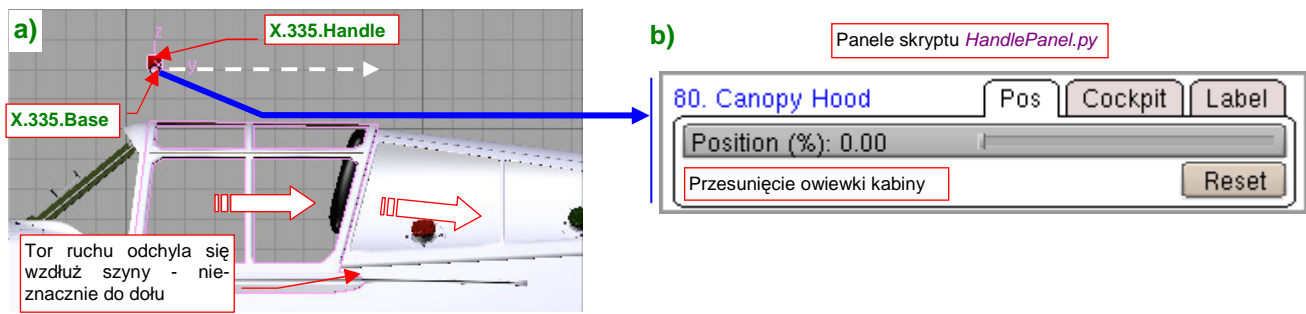
Było to dość dziwne rozwiązanie: w lewej szybie ponad „korkami” były wycięte dwa otwory (Rysunek 7.7.34):



Rysunek 7.7.34 Modyfikacja lewej tylnej szyby kabiny

Okrągłe wzmocnienia wokół „korków”, oraz umieszczone na nich śruby to wymodelowane siatki. Chciałem, by były widoczne, bo Blender ignoruje tekstury nierówności „za szybą” (por. Rysunek 7.7.34b) i Rysunek 7.7.33b).

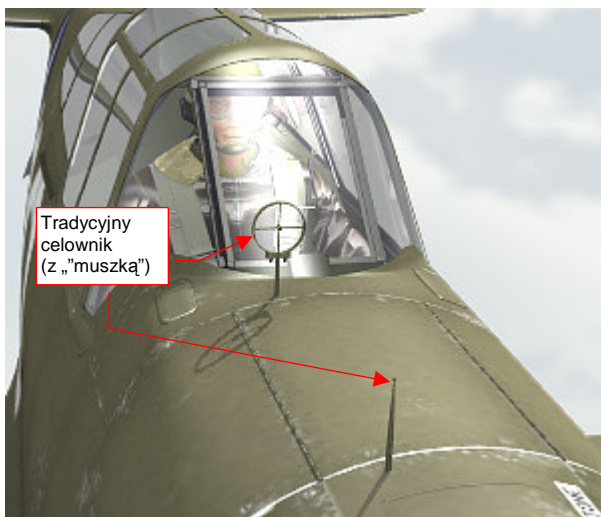
Proponuję dodać do naszego zestawu kontrolki sterujących modelem kolejną, odsuwającą owiewkę kabiny pilota (Rysunek 7.7.35) . W domyślnym położeniu (0%) jest zamknięta (Rysunek 7.7.35b):



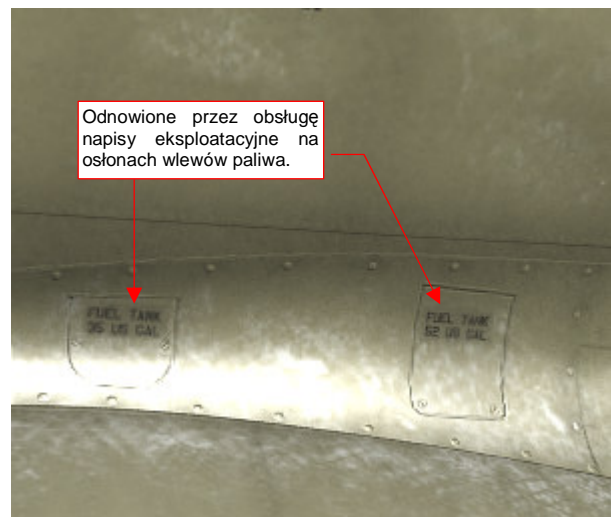
Rysunek 7.7.35 Nowa kontrolka, sterująca przesunięciem osłony kabiny

Szyny owiewki P-40 były nieznacznie wygięte do dołu (Rysunek 7.7.35a), stąd ruch owiewki jest powiązany z przesunięciem uchwytu (X.335.Handle) za pomocą dwóch ograniczeń Transformation. Jedno z nich wywołuje przesunięcie w poziomie, a drugie - to niewielki obrót (o 1°).

Obok celownika odbłaskowego, P-40 miały także drugi, tradycyjny. Umieszczono go z boku (Rysunek 7.7.36):

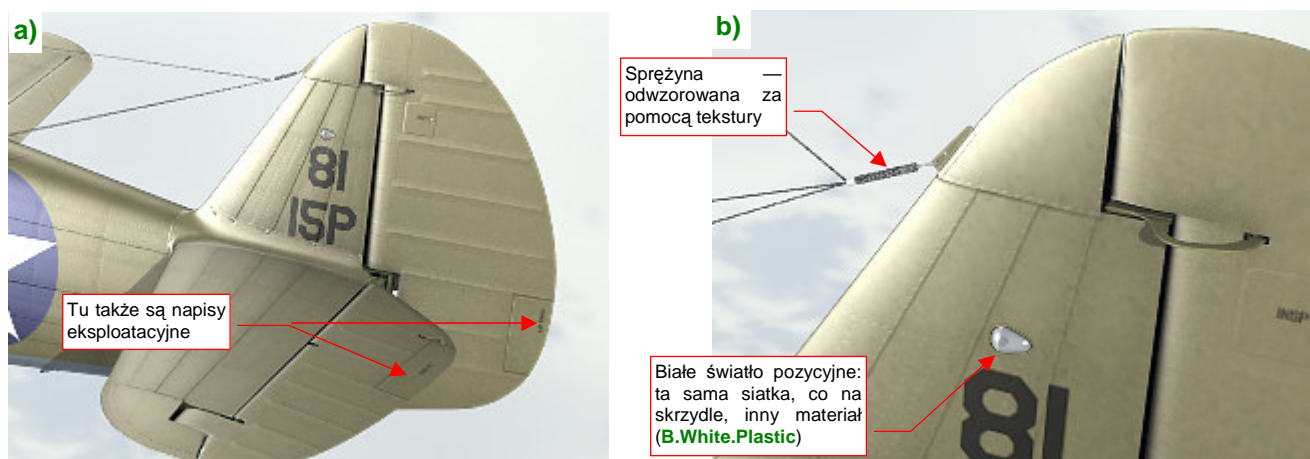


Rysunek 7.7.36 Dodatkowy celownik „szczerbinkowy”



Rysunek 7.7.37 Odnawiane napisy eksploatacyjne

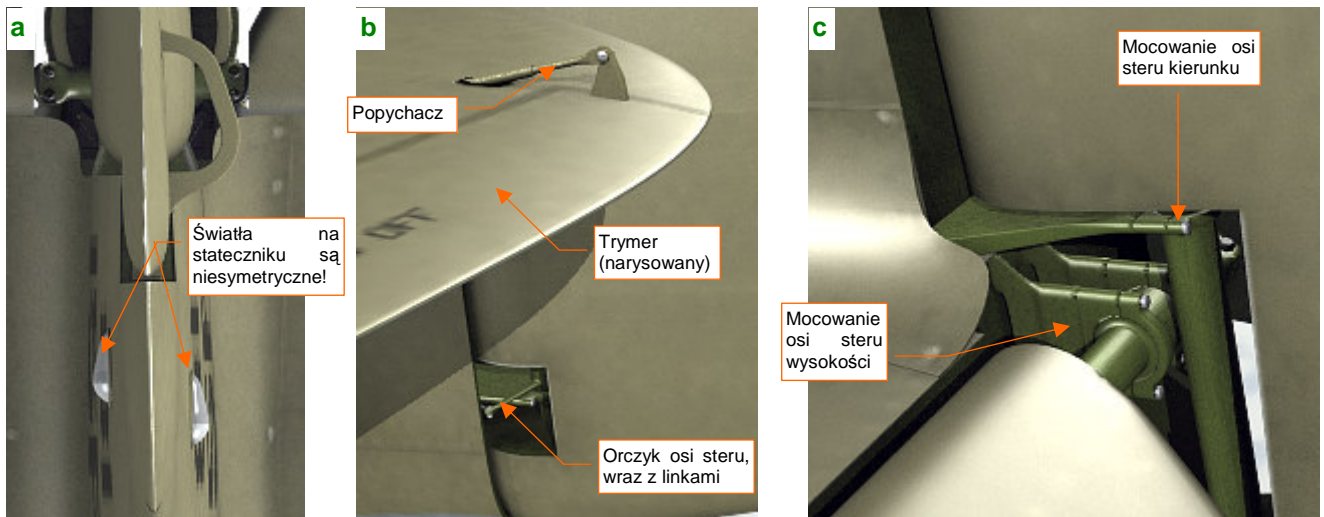
Nawet w AVG, mimo „polowych” warunków eksploatacji, mechanicy odnawiali co ważniejsze napisy eksploatacyjne. Naniósłem na tekstury teksty z osłon wlotów paliwa do zbiorników skrzydłowych (Rysunek 7.7.37). Inne napisy, które potwierdzają zdjęcia, umieściłem na usterzeniu (Rysunek 7.7.38a):



Rysunek 7.7.38 Usterzenie, wraz z anteną

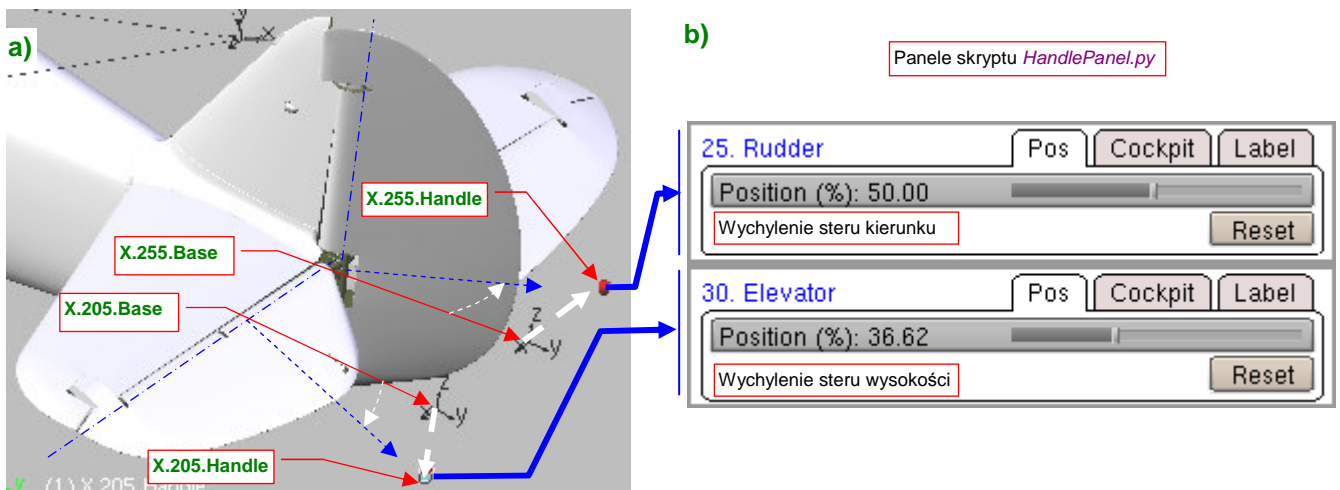
Na tym etapie prac wiele części masz już gotowych: na przykład światło na usterzeniu (Rysunek 7.7.38b).

Jeżeli chodzi o światła sygnalizacyjne, to rysunek fabryczny statecznika pionowego P-40 pozwolił mi wyśledzić pewien szczegół: były umieszczone niesymetrycznie (Rysunek 7.7.39a). Innym elementem „z odzysku” jest sprężyna, naciągająca druty anteny<sup>1</sup> (Rysunek 7.7.38b). Odwzorowałem ją za pomocą odpowiedniej tekstury przejrzystości. (To ten sam materiał **B.Steel.Spring**, którego użyliśmy w zespole kółka ogonowego — por. str. 429 i 432). Kolejnym przydatnym drobiazgiem okazały się popychacze, wykorzystane w klapach. Mogłem ich użyć powtórnie, wraz ze śrubką i nakrętką, do zamodelowania napędu trymeru (Rysunek 7.7.39b). W porównaniu z tym szczegółem sam trymer wygląda trochę nieładnie — bo jest tylko narysowany, a nie wycięty w siatce. Pozostawię go jednak już tak, jak jest. Z nieco większej odległości (por. np. Rysunek 7.7.38a) to zupełnie nie razi. Zawsze możesz sam uszczegółwić ten fragment.



Rysunek 7.7.39 Szczegóły usterzenia ogonowego

W przedniej części steru kierunku P-40 znajdowały się dwa otwory. W dolnym (Rysunek 7.7.39b) umieściłem orczyk na osi steru. Do końców orczyka przymocowane są linki. Do ich zamodelowania wykorzystałem tę samą siatkę, co w kabinie pilota (w istocie to końce tych samych linek). W górnym odwzorowałem okucia, mocujące osie sterów do stateczników (Rysunek 7.7.39c). Na koniec przygotowałem odpowiednie mechanizmy poruszania powierzchniami sterowymi (Rysunek 7.7.40) :



Rysunek 7.7.40 Nowe kontrolki, sterujące sterem kierunku i wysokości

<sup>1</sup> Swoją drogą, ta postać anteny była odziedziczona po P-36, i pochodziła z poprzedniej epoki rozwoju lotniczych urządzeń radiowych. Musiała stawiać poważny opór. Dla porównania: prototyp MiG-3 miał tylko jedną taką pochyłą „nitkę”, biegnącą od statecznika pionowego do osłony kabiny. Zastąpienie jej masztem z zupełnie poziomą anteną zwiększyło prędkość maksymalną o 3 km/h. A P-36 i P-40 B/C/D/E miały aż trzy takie druty, i to o wiele dłuższe! Myślę, że mogły spowalniać samolot nawet o jakieś 10 km/h. W P-40 maszt antenowy zaczęto stosować od wersji F.

W tej sekcji przebyliśmy drogę „od nosa do ogona” samolotu, uzupełniając po drodze wszystkie szczegóły zewnętrzne. Zakończyliśmy w ten sposób budowę naszego modelu!

Rysunek 7.7.41 przedstawia rezultat:



Rysunek 7.7.41 Gotowy model

Na razie nie umieszczam go na żadnym zdjęciu. (Jakoś nie pasuje mi to wysunięte podwozie do dotychczasowego tła, pokazującego ziemię z dużej wysokości). W następnym rozdziale pokażę, jak dopasować nasz model do jakiegoś przekonującego otoczenia.



### Podsumownie

- Aby uzyskać w zbliżeniach modelu bardzo wyraźne napisy i oznaczenia, możesz użyć tekstur wektorowych (str. 491). Wymaga to jednak zastosowania specjalnej wtyczki do Blendera, która obecnie działa tylko w środowisku Windows. Jest to poważna przeszkoda, gdy np. chciałbyś taki model udostępnić komuś, kto używa innego systemu operacyjnego;
- Praca nad detalami oznacza także konieczność poprawiania fragmentów tekstur. Może to być, na przykład, tekstura barwy (str. 504), czy nierówności (str. 503);
- Kolektor spalin najprościej jest wykonać z pojedynczej, prostej „rury wydechowej”, stosując kilka modyfikatorów. Modyfikator *Curve Deform* użyłem do wygięcia siatki w odpowiedni kształt (str. 492). Potem powiększyłem siatkę pojedynczej rury za pomocą modyfikatorów *Array* (str. 493);
- Ściany chłodnic cieczy można odwzorować za pomocą odpowiedniej tekstury przejrzystości (str. 494);
- W czasie prac nad ostatnimi szczegółami może się zawsze pojawić konieczność poprawy jakiegoś dawno uformowanego fragmentu poszycia. Tak się stało z otworami wyrzutników łusek karabinowych, które umieściłem na spodzie kadłuba (str. 496). Takie poprawki są dość kłopotliwe, gdyż wymagają zmian w rozwinęciach UV, a te z kolei pociągają za sobą konieczność modyfikacji podstawowych tekstur;
- Do paneli sterujących modelem (skrypt *HandlePanel.py*) warto dodać także uchwyt odpowiedzialny za obrót śmigła (str. 496). To ułatwi „pozowanie” modelu na ziemi i uzyskanie efektu rozmycia łopat w locie;
- Otwory w osłonach luf karabinów maszynowych najłatwiej jest uzyskać za pomocą tekstury przejrzystości (str. 497);
- Unikaj pozostawienia jednego z płatów z ujemnym współczynnikiem skali (a taki pozostaje np. po lustrzanym odbiciu — transformacji *Mirror*). Jakakolwiek negatywna liczba występująca w polach *ScaleX*, *Y* lub *Z* obiektu powoduje, że wiele ograniczeń, w tym *Locked Track*, nie działa poprawnie. Ten „feler” propaguje się także hierarchicznie — na wszystkie bezpośrednie obiekty potomne (*immediate children*). Tak właśnie było z osiami klap i lotek prawego skrzydła w moim modelu (str. 499, 500). Konieczne okazało się przypisanie im nowego, „pozytywnego rodzica”;
- Do zestawu paneli sterowania modelem warto dodać, oprócz sterów kierunku, lotek i klap, także owiewkę kabiny (str. 504);

## **Rozdział 8. Rendering**

Pokazuję tu ostateczne zakończenie pracy nad naszym modelem: umieszczam go na jakimś przekonującym obrazku.

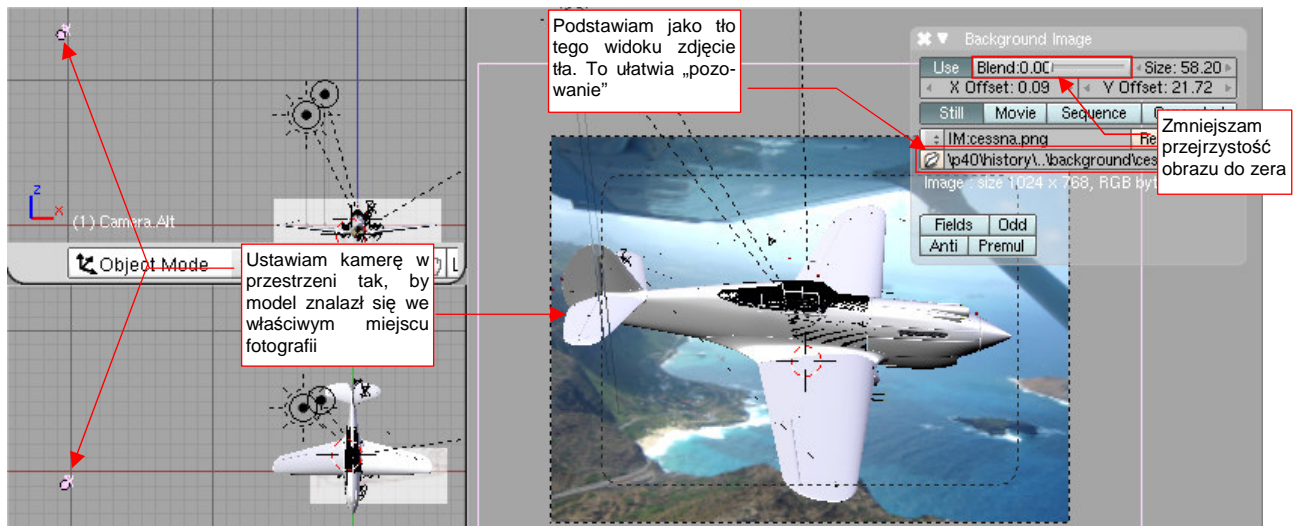
Ten rozdział zawiera tylko dwie sekcje. Pierwsza sekcja opisuje zagadnienia związane z przygotowaniem sceny w locie. Jak się przekonasz, większość czasu będziesz się w niej zajmował „kręceniem śmigłem”. (Bo uzyskanie poprawnego i szybkiego obrazu rozmytych ruchem łopat nie jest takie proste).

Druga sekcja pokazuje, jak „postawić samolot na ziemi”, którą widać na podstawionej za modelem fotografii. W tym przypadku większość wysiłku idzie na „walkę z cieniem” — samolotu, oczywiście. (Pokażę, jakimi zabiegami można dobrze scalić komputerowy model z obrazem tła).

## 8.1 Realistyczna scena w locie

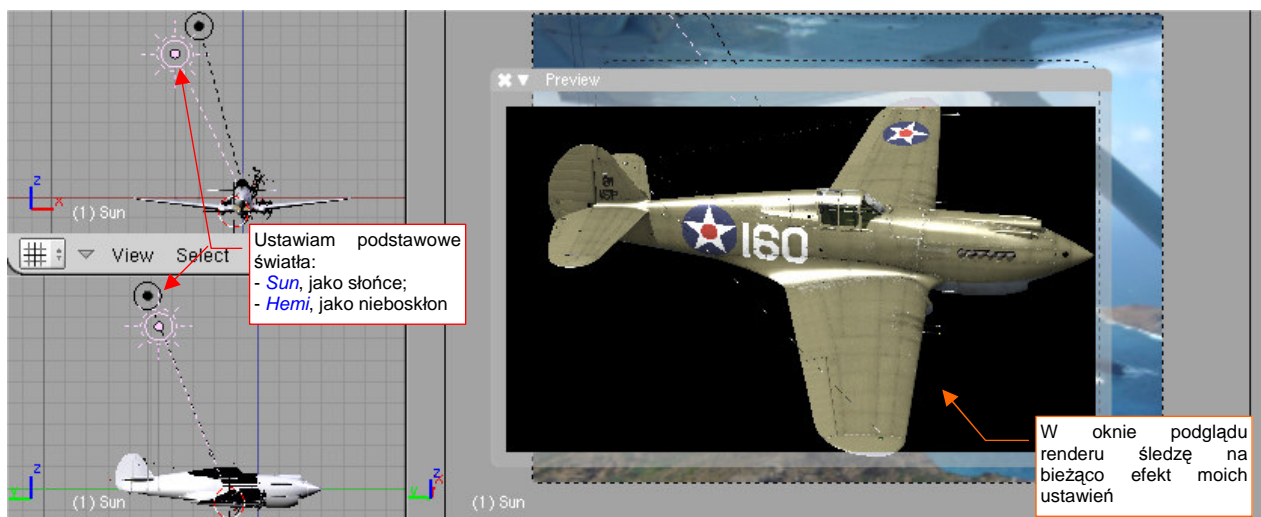
W tej sekcji pokażę, jak przygotować scenę w locie. Omówię poszczególne zagadnienia, posługując się przykładami z różnych scen<sup>1</sup>. Zaczynam od wyszukania w Internecie jakiegoś darmowego zdjęcia o odpowiedniej rozdzielczości. Potem podstawiam je za modelem, komponując prostą scenę. Jest to dla mnie swoisty „ostateczny test” realizmu: jeżeli mój samolot będzie dobrze wyglądał na tle fotografii, oznacza to że udało mi się go dobrze odwzorować. Specjalnie zresztą wyszukuję zdjęcia okolic, nad którymi naprawdę latała konkretna maszyna, której barwy nosi mój model. Chcę w ten sposób odtworzyć jej „naturalne otoczenie”.

Pracę nad sceną zaczynam od podstawienia „pod” widok z kamery fotografii — jako tła (*Background Image*<sup>2</sup>) (Rysunek 8.1.1). Potem w widokach z przodu i z góry przesuwam odpowiednio kamerę, „pozując” model:



Rysunek 8.1.1 „Pozowanie” — dopasowanie pozycji kamery (a w efekcie – modelu) do tła

Następnie włączam podgląd renderu (**Shift-P**) i ustalam pozycje i energie podstawowych „lamp” sceny: **Sun** i **Hemi.Upper**. Robię to tak, by światło padało na model z tej samej strony, co na fotografii (Rysunek 8.1.2):



Rysunek 8.1.2 Dopasowanie położenia światła — słońca i nieboskłonu

<sup>1</sup> W związku z tym, trochę nietypowo, w folderze *p40history* znajdziesz nie jeden, a kilka plików Blendera związanych z tą sekcją książki (*P40B-8.01a.blend...p40B-8.01d.blend*). Każdy z nich to pojedyncza scena, z innym tłem. Źródła zdjęć, które wykorzystałem jako tło, podaję w podpisach pod ilustracją z ostatecznym renderem każdej sceny.

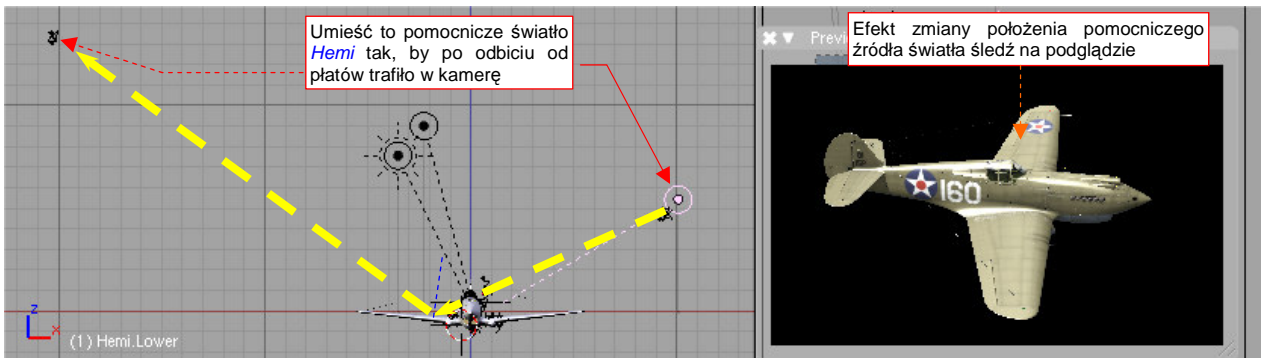
<sup>2</sup> Por. str. 772. Rozmiary i przesunięcia *X Offset* i *Y Offset* obrazu w projekcji „z kamery” nie mają znaczenia. Zdjęcie możesz więc podstawiać wprost w miejsce planów samolotu. (A potem bez problemu znów na to miejsce wstawić plan).

Gdy dopasujesz oświetlenie, przypisz jeszcze obraz jako tło sceny (por. str. 971). Potem wykonaj pierwszy render i sprawdź, jak się na nim prezentuje model (Rysunek 8.1.3):



Rysunek 8.1.3 Próbny render

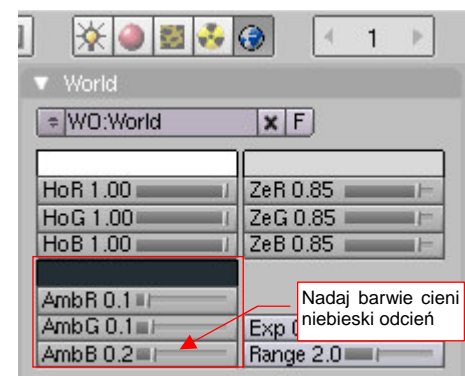
Płaty wyglądają „płasko”. Ożywimy je za pomocą słabego ( $Energy = 0.2.. 0.3$ ) światła *Hemi* (Rysunek 8.1.4):



Rysunek 8.1.4 Pomocnicze światło — odbłask na płatach i usterzeniu poziomym

Nawet za pomocą słabej, pomocniczej lampy *Hemi*, można uzyskać silny odbłask. Należy ustawić ją tak, by światło odbiło się od powierzchni modelu i trafiło w obiektyw kamery (Rysunek 8.1.4). (Pamiętasz lekcje fizyki? „Kąt padania równa się kątowi odbicia”. Tutaj także!). W takim odbłasku stają się wyraźne wszystkie łączenia blach i drobne nierówności. Dzięki nim cały model zaczyna wyglądać ciekawiej. To pomocnicze światło możesz umieścić w dowolnym miejscu — byleby wywołało na modelu odpowiedni efekt.

Inną adaptacją to kolor cieni (a dokładniej: barwa bezkierunkowego światła otaczającego: *Ambient*). Do tej pory stosowaliśmy neutralny szary. Scena na zdjęciu sprawia jednak wrażenie niebieskawej. Zmień barwę światła otaczającego nieco bardziej niebieską, by lepiej wpasować model w otoczenie (Rysunek 8.1.5).



Rysunek 8.1.5 Zmiana barwy cieni — z 10% szarości na niewiele jaśniejszą szarobłękitną

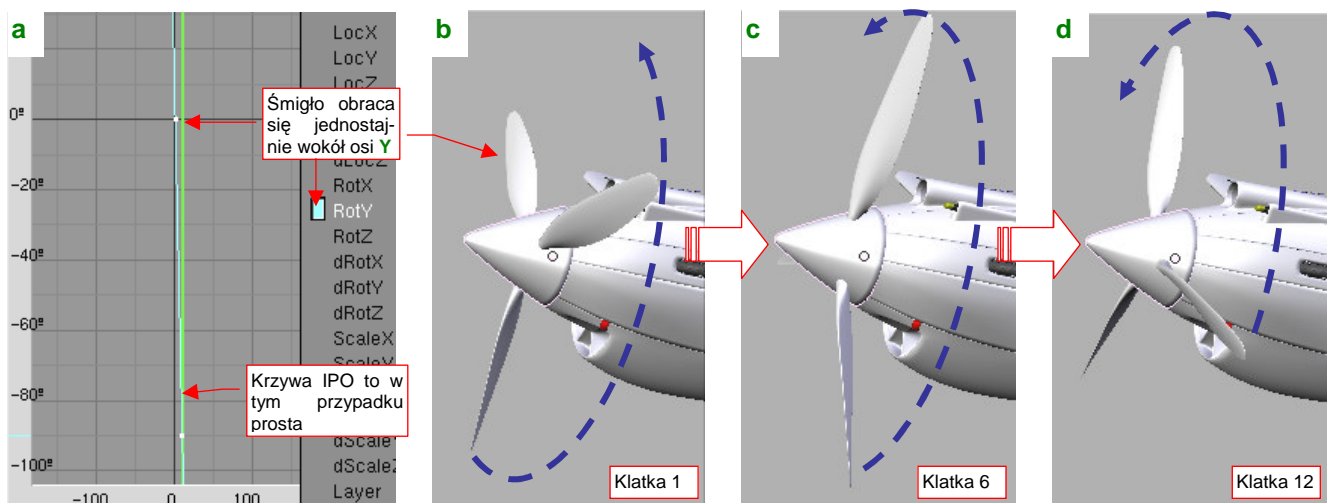
Po tych poprawkach czas na kolejny testowy render (Rysunek 8.1.6). Uważam, że tym razem oświetlenie wygląda już całkiem poprawnie (porównaj je zresztą z wersją pokazywaną przez Rysunek 8.1.3):



Rysunek 8.1.6 Próbny render — wersja ze światłem pomocniczym i lekko niebieskawymi cieniami (**Ambient**)

Do takich „przymiarek” modelu wyłączaj warstwy z elementami które nie są na razie istotne, a wydłużyłyby czas renderowania. (Jest to zazwyczaj pilot i wnętrze kabiny). Na razie na renderze nie są także widoczne łopaty śmigła, a sam kołpak jest nieruchomy. Czas już, by zaczęły się kręcić!

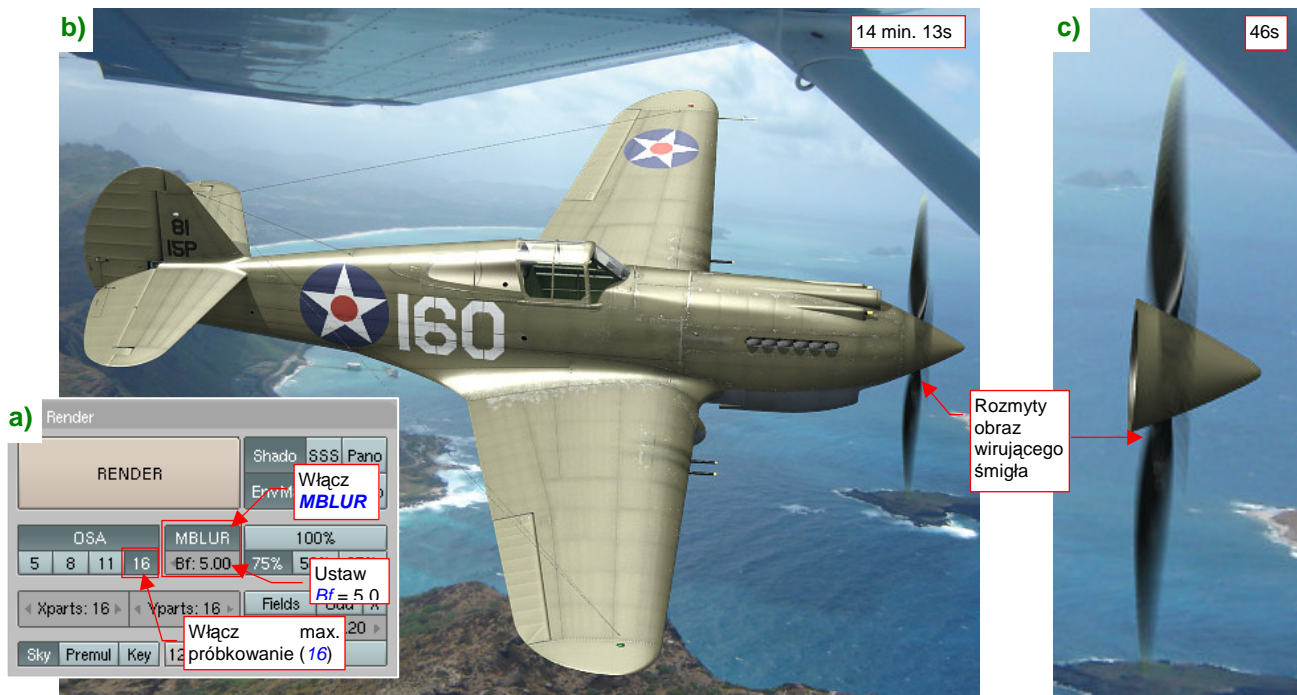
Mógłbym przypisać sekwencję ruchu (tzw. „krzywą IPO” — *IPO curve*) do uchwytu, poruszającego kołpakiem. Jednak z powodu pewnych problemów, które wystąpią za chwilę, wolę zmniejszyć do zera jego wpływ na przypisany obiekt. Odpowiednią krzywą IPO, odpowiedzialną za obrót śmigła, przypisuję bezpośrednio do kołpaka (Rysunek 8.1.7a). Wraz z nim obracają się łopaty. (Naciśnij **Alt-A** w którychś z okien **3D View**, by się o tym przekonać — Rysunek 8.1.7b,c,d):



Rysunek 8.1.7 Animacja obrotu śmigła

Szczegółowy opis, jak przypisać śmigłu linię ruchu, znajdziesz na str. 1029 i następnych. Śmigła samolotów z lat 40-tych obracały się zazwyczaj z prędkością 1000 – 1500 obr/min. (Reduktory silników miały przełożenie ok. 2:1). Tej prędkości odpowiada w naszym modelu obrót o 60 – 90° pomiędzy klatkami 1 i 16.

Gdy animacja śmigła jest już gotowa, możemy ten ruch uchwycić na naszej scenie. Wystarczy w panelu **Render** włączyć opcję **MBLUR** (*Motion Blur* — rozmycie ruchu). Aby obraz wirujących łopat śmigła był bardzo znacznie rozmyty, przestaw parametr **OSA** na **16**, a współczynnik **Bf** (*Blur factor*) na wartość maksymalną: 5.0 (Rysunek 8.1.8a). Potem wystarczy wcisnąć przycisk **Render**, by otrzymać rezultat (Rysunek 8.1.8b):



Rysunek 8.1.8 Efekt „rozmycia” wirujących łopat śmigła — metoda najprostsza (i najwolniejsza)

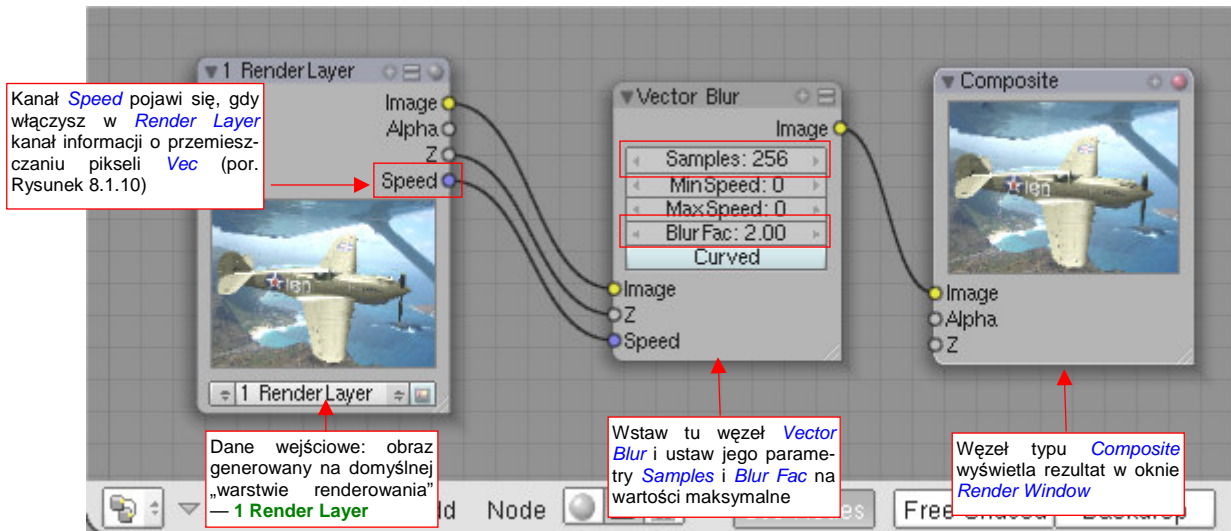
Efekt jest całkiem zadowalający tylko...strasznie długo się generował! Czas tworzenia ostatecznego obrazu wydłużył się z ok. 45s (**MBLUR** wyłączone) do 14 min 13s (**MBLUR** włączone). Patrząc na okno renderu w czasie tego kwadransa, zapewne zauważyłeś, że Blender generował nie jeden, a 16 kolejnych obrazów! (Dokładnie rzecz biorąc: jest ich tyle, na ile jest ustawiona liczba próbek **OSA**). Było to 16 kolejnych klatek animacji. W naszym przypadku te ujęcia różnią się wyłącznie pozycją kołpaka i łopat śmigła<sup>1</sup>. Potem te obrazy zostały połączone, tworząc znany z fotografii efekt „rozmycia ruchu”.

Jeżeli tak długi czas generowania ostatecznego obrazu Ci nie przeszkadza, możesz nie czytać następnych stron i pozostać przy prostym użyciu **MBLUR**. Testowe renderowanie można przecież zawsze wykonać, bez tej opcji. Włączysz ją dopiero wtedy, gdy będziesz generował ostateczny rezultat. Wszystko właściwie zależy od rozmiaru (w pikselach) obrazu, który tworzysz, i czasu, jaki to zajmuje Twojemu komputerowi.

Osobiście jednak nie lubię, gdy renderowanie trwa zbyt długo. Zawsze na finalnym obrazie też znajdę jakiś błąd, który koniecznie muszę poprawić. Koryguję go, i zaraz znów zapuszczam kolejne „finalne” renderowanie. Oczekiwanie za każdym razem 15min, patrząc jak w tym czasie komputer niepotrzebnie generuje 16 razy to samo, to dla mnie za dużo! Popatrz na Rysunek 8.1.8c). Renderowanie z **MBLUR** tylko tego, co naprawdę się zmienia — samego śmigła — trwa raptem 46s. Reszta to czysta strata czasu!

<sup>1</sup> Dlatego przy wzmiance o animacji (str. 511) podałem, że prędkości obrotowej 1500obr/min odpowiada obrót śmigła o 90° pomiędzy klatkami 1 i 16. Możesz traktować ustawioną w **OSA** liczbę próbek jako „czas wykonania fotografii” naszego modelu. (Nie znam się na fotografii, ale to się chyba fachowo nazywa „przesłona”, albo „czas ekspozycji”). Prędkość 1500 obr/min to 25 obr/s. Gdybyś wykonywał zdjęcie tego samolotu z przesłoną 1/100s, w ciągu tego czasu łopata śmigła wykona 25 obr/s \* 1/100s = 0.25 pełnego obrotu, czyli właśnie 90°. Śmigła większości samolotów obracały się trochę wolniej. Przy tej samej przesłonie i prędkości obrotowej 1000 obr/min łopata zatoczy proporcjonalnie mniejszy łuk — 60°. Oczywiście nie jestem pewien, czy przypadkiem fotografowie z lat 40-tych nie ustawiali przesłony na krótsze czasy — np. 1/200s. Wtedy ślady śmigieł byłyby dwa razy węższe (i odpowiednio „gęstsze”). Wydaje mi się jednak, że takie i jeszcze krótsze przesłony zaczęły się pojawiać w aparatach fotograficznych w latach 1944-1945. To chyba sprawa postępu technicznego. Gdy przeglądam zdjęcia dwupłatów z lat 20-tych z pracującym silnikiem, ślad ich śmigieł jest kołem. Oznacza to, że do wykonania tych fotografii użyto dłuższych czasów ekspozycji — być może 1/50, a może nawet 1/25s. Chyba innych aparatów po prostu wtedy nie było....

Czy można uzyskać efekt rozmycia ruchu jakoś szybciej, by nie czekać kwadransa na każdy render? Spróbujmy innej metody: *Vector Blur*. Jest to efekt, który można uzyskać po przełączeniu sterowania renderowaniem na tzw. „węzły kompozycji” (*Composite Nodes* — Rysunek 8.1.9):

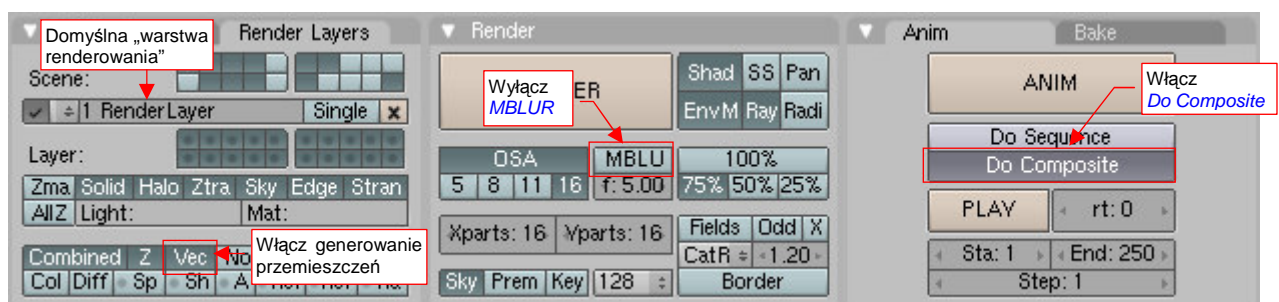


Rysunek 8.1.9 Przygotowanie innego efektu rozmycia — *Vector Blur*

Na pierwszy rzut oka sterowanie efektem za pomocą węzłów może wyglądać trochę obco. Myślę jednak, że szybko się do niego przekonasz. Może w paru słowach poniżej opiszę, jak należy „czytać” takie schematy.

Węzły to prostokąty połączone liniami. Każda linia oznacza przepływ jakichś danych. Każdy węzeł po prawej stronie ma kanały wyjściowe: to dane, które generuje. (Na przykład węzeł *Composite* nie ma żadnych, bo to *output node*: wyświetla otrzymany obraz w oknie na ekranie). Po prawej stronie węzła znajdują się „wtyczki” na oczekiwane dane wejściowe. (Tu z kolei węzeł *Render Layer* nie ma żadnych, bo to *input node*: dostarcza rezultatów działania podstawowego renderera sceny). Nie zawsze musisz podłączać wszystkie „wtyczki” węzła — to zależy od jego specyfiki. (Czasami niektóre informacje mogą być pominięte). Najprostsza kompozycja tworząca render sceny składa się z dwóch węzłów: *Render Layer*, dostarczającego danych, oraz *Composite*, wyświetlającego otrzymany obraz w *Render Window* Blendera. To połączenie dostarcza tego samego rezultatu co renderowanie bez użycia węzłów. Istota wykorzystania tego narzędzia polega na tym, że pomiędzy te dwa domyślne węzły możesz wstawić dziesiątki innych. Pozwala to uzyskiwać różnorodne efekty, niedostępne w żaden inny sposób. Dokładniejsze wprowadzenie w ten temat znajdziesz na str. 1033.

W układzie, który pokazuje Rysunek 8.1.9, pomiędzy węzły *Render Layer* i *Composite* wstaw efekt dodatkowy: *Vector Blur*. (Oczywiście, przełączając odpowiednie „kabelki”). *Vector Blur* musi otrzymać na wejściu, oprócz pikseli obrazu (wtyczka *Image*) i ich głębokości (wtyczka *Z*) także informację o przemieszczeniach: (wtyczka *Speed*). Ten kanał pojawi się wśród danych wejściowych, gdy włączysz dla domyślnej warstwy renderowania kanał *Vec* w panelu *Render Layers* — Rysunek 8.1.10):



Rysunek 8.1.10 Ustawienia opcji renderera do współpracy z efektem *Vector Blur*

Wyłącz także, w panelu *Render*, opcję *MBLUR*. Włącz za to w panelu *Anim* opcję *Do Composite*. Inaczej Blender nie zwróci uwagi na nasze węzły.

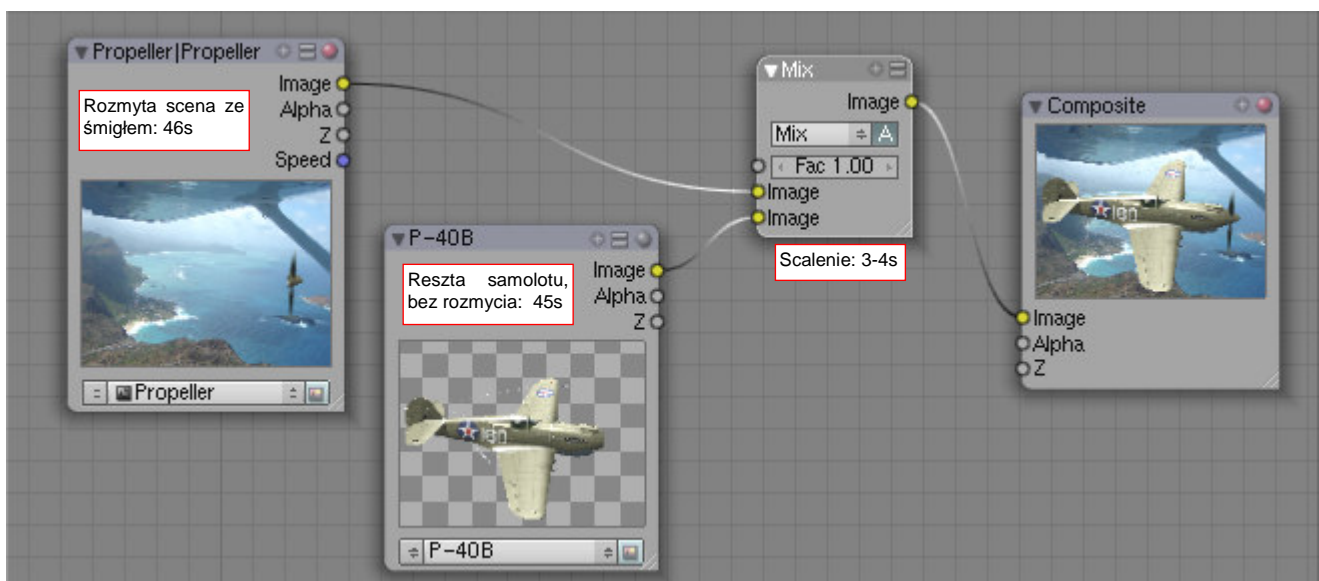
Rezultat użycia węzła *Vector Blur* przedstawia Rysunek 8.1.11:



Rysunek 8.1.11 Rezultat zastosowania *Vector Blur*

Czas generowania tego obrazu różni się tylko o kilkanaście sekund od wersji bez śmigła i wynosi 58s. Ale to koniec dobrych wiadomości. Rozmycie jest wyraźnie słabsze niż to, uzyskane za pomocą *Motion Blur* (por. str. 512, Rysunek 8.1.8). W dodatku na kołpaku pojawił się wyraźny cień nieruchomej łopaty śmigła. I nie jest to jakiś drobny błąd, bo w opisie działania *Vector Blur* wyraźnie zaznaczono, że cienie nie ulegają rozmyciu. Krótko mówiąc: efekt *Vector Blur* nie nadaje się do tego przypadku. Mimo ustawienia maksymalnej liczby próbek (*Samples* = 256) i współczynnika rozmycia (*Blur Fac* = 2.0) efekt jest daleki od oczekiwań.

Czy jest jakiś inny sposób, który pozwoli nam uniknąć renderowania 16 razy tego samego obrazu samolotu? Pewien pomysł nasunął mi stosunkowo krótki czas renderowania z *MBLUR* samego śmigła (por. Rysunek 8.1.8c). A gdyby tak podzielić samolot na dwie sceny? Na jednej wygenerować obraz samych wirujących części, rozmytych za pomocą *Motion Blur*. Na drugiej renderować całą resztę samolotu, bez rozmycia. A potem te dwa wyniki scalić za pomocą odpowiedniego węzła? Tak, to jak najbardziej jest możliwe! (Rysunek 8.1.12):



Rysunek 8.1.12 Kompozycja dwóch scen: rozmytego obrotu śmigła (*Propeller*) i reszty samolotu (*P-40B*).



Przygotuj drugą, pomocniczą scenę i nadaj jej nazwę — powiedzmy, **Propeller**. Umieść na niej referencje do aktualnych światła i kamer, oraz wirujące elementy śmigła. Włącz na niej **MBLUR**. Ta scena tworzy swój render „tradycyjnie”, bez węzłów. Na scenie podstawowej (**P-40B**) wyłącz widoczność warstwy śmigła. Przygotuj jej węzły kompozycji tak, jak pokazuje to Rysunek 8.1.12 (str. 514). Rezultat pokazuje Rysunek 8.1.13:



Rysunek 8.1.13 Ostateczny rendering — rezultat złożenia scen (tło: [www.archerbravo.com](http://www.archerbravo.com))

(Szczegółowy opis przygotowania tego rozwiązania znajdziesz na str. 779 i następnych). Wydaje się, że ta kombinacja pozwala uzyskać dziesięć razy krótszy czas renderowania od „zwykłego” MBLUR. Dla zestawu warstw, z którymi porównywaliśmy się do tej pory (czyli bez pilota i wnętrza kabiny) generuje się w 1 min. 35s. (16 renderów śmigła i połączenie ich w MBLUR zajęło tyle samo czasu, co render reszty samolotu — por. Rysunek 8.1.12). Ostateczny render (Rysunek 8.1.13), na którym włączyłem wszystkie warstwy, trwał oczywiście nieco dłużej: 2 min. 15s. W każdym razie są to czasy całkiem rozsądne, więc ta metoda „dwóch scen” jest ze wszelkich miar godna polecenia. (Zwróć uwagę, że nie można jej osiągnąć inaczej, jak przez montaż za pomocą węzłów. I za to te **Composite Nodes** lubię!)

Ubočnym efektem nałożenia samolotu bez tła na obraz z tłem jest uzyskanie za szybami kabiny poprawnego fragmentu podstawionej fotografii (Rysunek 8.1.14). Były z tym problemy dla szyb stosujących metody **Ray Transp** (por str. 981, Rysunek 17.12.13):



Rysunek 8.1.14 Efekt „zanikania szyby” przy złożeniu scen

Niestety, przy okazji znikły z szyb odbłyski, które powinno rzucać słońce — więc nie jest to dobry efekt! Dlatego w ostatecznej scenie (Rysunek 8.1.13) model jest bardziej przechylony (por. Rysunek 8.1.14). Chodziło o to, by nawet przy zamkniętej kabinie jej szkło znalazło się na tle samolotu. Wtedy odbłask słońca nie znika.

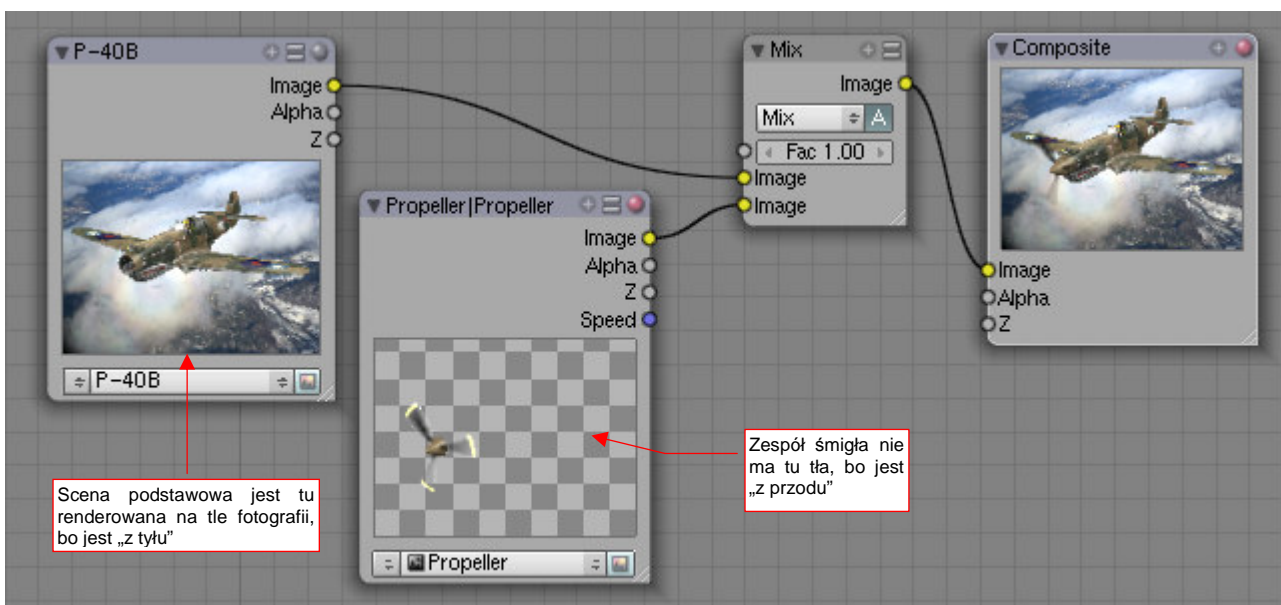
Fotografię tła należy umieścić zawsze na tej scenie, która jest dalej od obserwatora. W kompozycji, nad którą pracowaliśmy na poprzednich stronach (np. str. 515 Rysunek 8.1.13) model był do nas ustawiony bokiem. Jednak jego nos minimalnie odchyłał się na zewnątrz (to może być zresztą efekt zniekształcenia perspektywicznego). O takim odchyleniu świadczy widoczne na pomocniczej scenie wnętrze kołpaka śmigła (por. chociażby na str. 512, Rysunek 8.1.8c). Dlatego na schemacie, który zastosowaliśmy, fotografia była podstawiona jako tło pomocniczej sceny **Propeller**, a reszta samolotu była przygotowywana na tle przezroczystym. W tej sytuacji mogliśmy się natknąć na opisany przed chwilą „efekt zanikania szyby”.

Na szczęście na wielu typowych scenach śmigło jest — z punktu widzenia obserwatora — przed samolotem. Popatrz chociażby na obraz, który pokazuje Rysunek 8.1.15:



Rysunek 8.1.15 W tej scenie śmigło znajduje się „przed” modelem (tło: © Tomo Yun, [www.yunphpto.net](http://www.yunphpto.net))

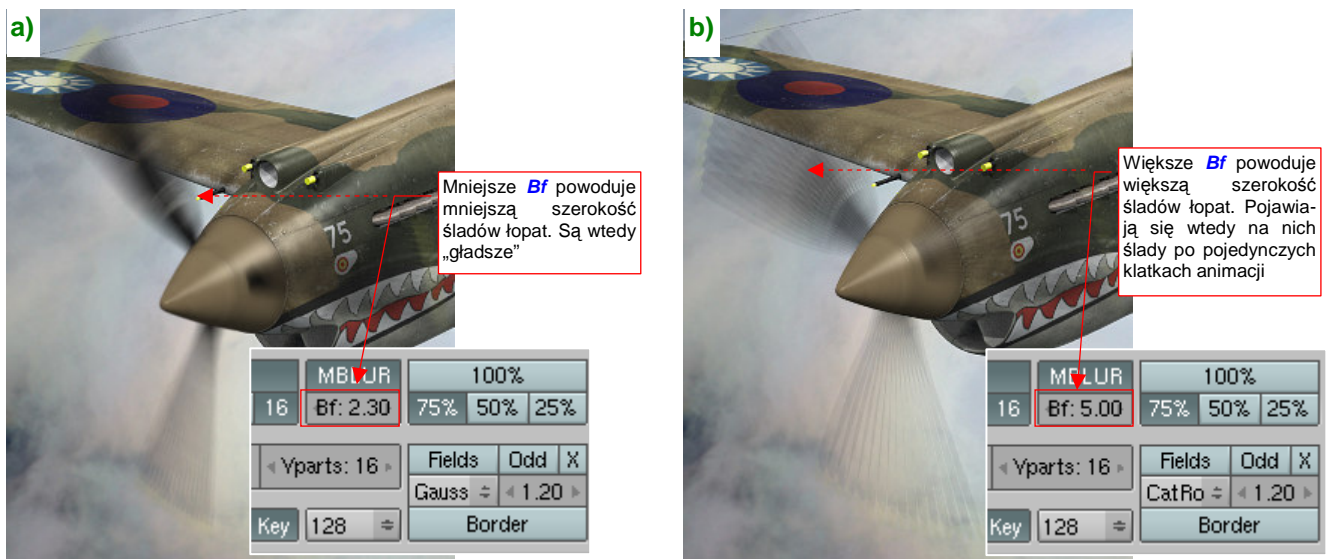
To inne zdjęcie i inna maszyna — „biała 75” W.N. Reeda z 3 dywizjonu AVG („Hell’s Angels”), uchwycona nad wyższymi partiami gór w okolicach Kunmingu. Obserwator widzi tutaj „portretowany” samolot nieco z przodu — pod kątem zbliżonym do 45°. W każdym razie, w tej kompozycji to scena z samolotem jest renderowana od razu na tle zdjęcia. Scena z wirującym śmigłem nie ma tu tła (Rysunek 8.1.16):



Rysunek 8.1.16 Kompozycja rezultatu — tym razem przejrzyste tło jest za śmigłem.

W tym układzie nie ma problemu z odbiaskami na szkłe kabiny (tylko znów trzeba ją pozować na jasnym tle).

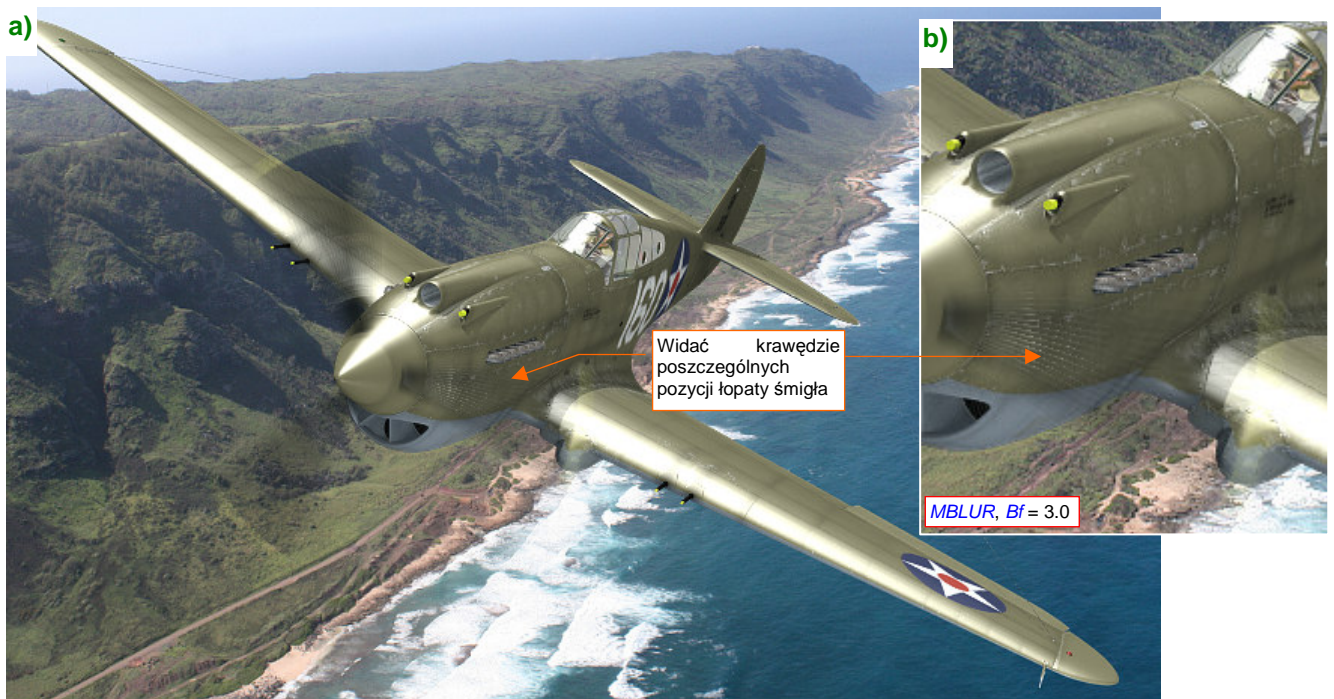
Dla tego samolotu z AVG zastosowałem mniejszą wartość współczynnika rozmycia —  $Bf$  — niż w poprzedniej kompozycji. Użyłem tu  $Bf = 2.3$ , by uzyskać płynne rozmycia łopat śmigła (Rysunek 8.1.17a):



Rysunek 8.1.17 Rozmycie śmigła dla różnych wartości  $Bf$  (Blur factor)

Takie śmigło wygląda, jak na zdjęciu zrobionym aparatem o krótkiej przesłonie ( $1/200s$  lub mniej). Gdybym pozostał przy  $MBLUR$  z  $Bf = 5.0$ , ślady po łopatach byłyby szersze. Dostrzegłbyś jednak na nich zarysy pozycji śmigła w poszczególnych klatkach, użytych do tego efektu (Rysunek 8.1.17b). To dyskwalifikuje takie ustawienie. Na szczęście do tej sceny tak wąskie ślady łopat są całkiem odpowiednie, bo ten samolot raczej nie leci na pełnych obrotach. (Popatrz na Rysunek 8.1.15 — owiewka kabiny jest otwarta).

Niestety, takie „twarde ślady” obrazów kolejnych klatek mogą się pojawiać także dla o wiele mniejszych wartości  $Bf$ . Wszystko zależy od oświetlenia. Jeżeli na krawędzi łopata śmigła pojawia się jasny odbłask słońca, to takie linie będą się wyróżniać na rozmyciu  $MBLUR$  nawet dla  $Bf = 3.0$  (Rysunek 8.1.18):

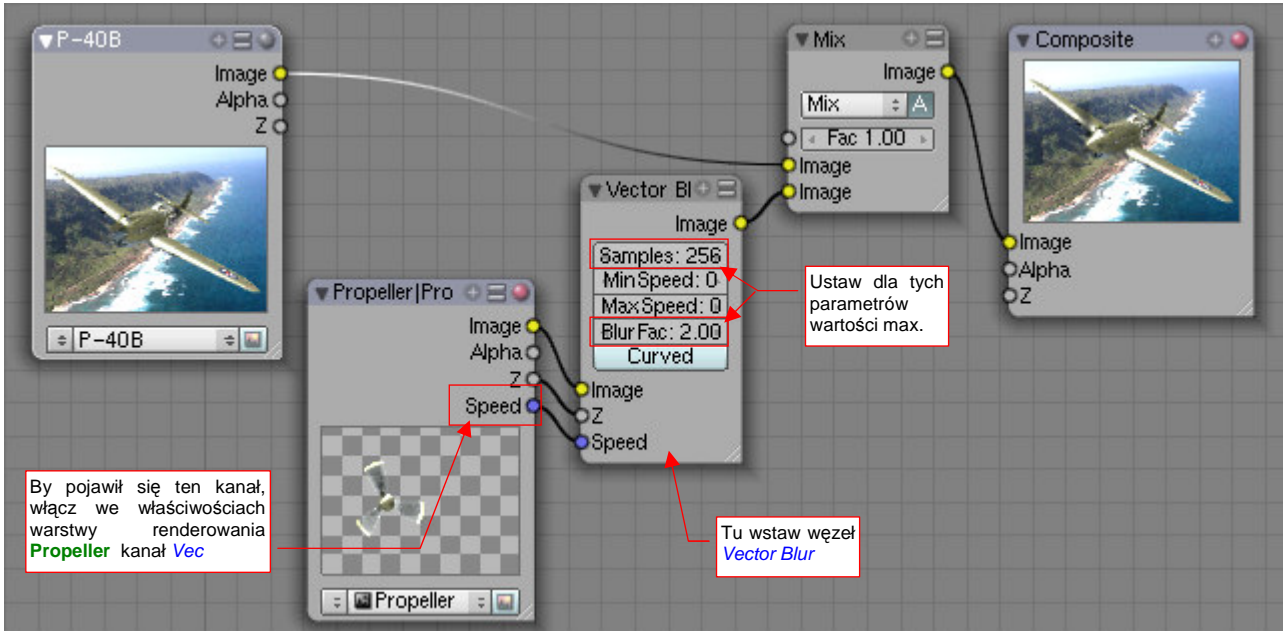


Rysunek 8.1.18 „Twarde” krawędzie poszczególnych pozycji śmigła dla rozmycia  $MBLUR$

Samolot na tej scenie to wznoszący się P-40B z 47 PS, stacjonującego w grudniu 1941 na Hawajach. Podczas wznoszenia obroty silnika są większe, więc łopaty śmigła zataczają szersze łuki. Niestety, pojawiły się na nich ostre krawędzie. (Popatrz na Rysunek 8.1.18b) — to fragment obrazu w rozmiarze oryginalnym).

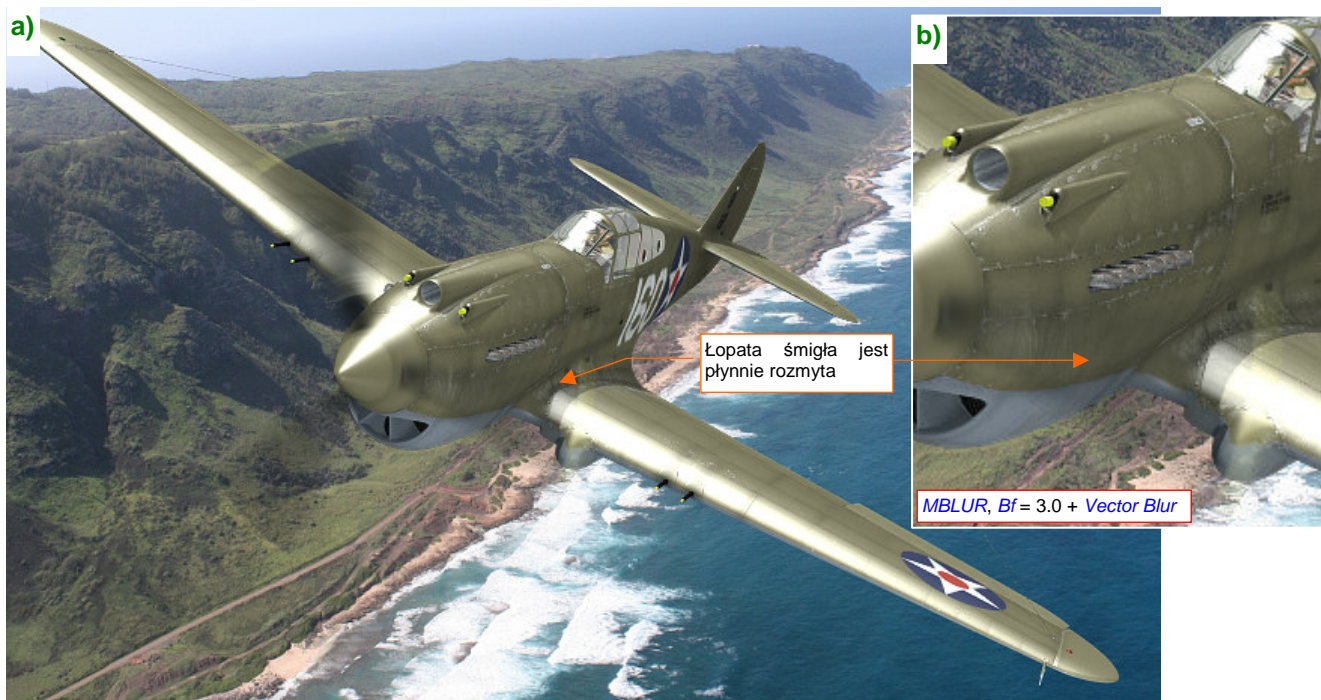
Czy coś można na takie ostre krawędzie zaradzić? W tej scenie silnik samolotu powinien pracować na wysokich obrotach, więc szersze rozmycia łopat śmigieł są bardzo potrzebne!

Oczywiście! Pamiętasz efekt *Vector Blur*? Gdy działał w pojedynkę, był za słaby. Ale kto powiedział, że nie można nim dodatkowo „rozmyć” efektów działania *Motion Blur (MBLUR)*!? Spróbujmy dać mu drugą szansę, i dodajmy ten węzeł do schematu (Rysunek 8.1.19):



Rysunek 8.1.19 Dodatkowe „rozmycie” śmigła za pomocą *Vector Blur*.

Włącz na warstwie renderowania **Propeller** pomocniczej sceny generowanie informacji o przesunięciach (przełącznik *Vec* — por str. 513, Rysunek 8.1.10). Dzięki temu w węźle reprezentującym tę warstwę pojawi się dodatkowy kanał *Speed* (Rysunek 8.1.19). Następnie wstaw do schematu nowy *Vector Blur*. Przyłącz jego wejścia do sceny **Propeller**, a wyjścia — do złożenia *Mix*. Zwiększ także współczynniki *Vector Blur* (*Samples* i *Blur Fac*) do maksimum (Rysunek 8.1.19). Rezultat przedstawia Rysunek 8.1.20:



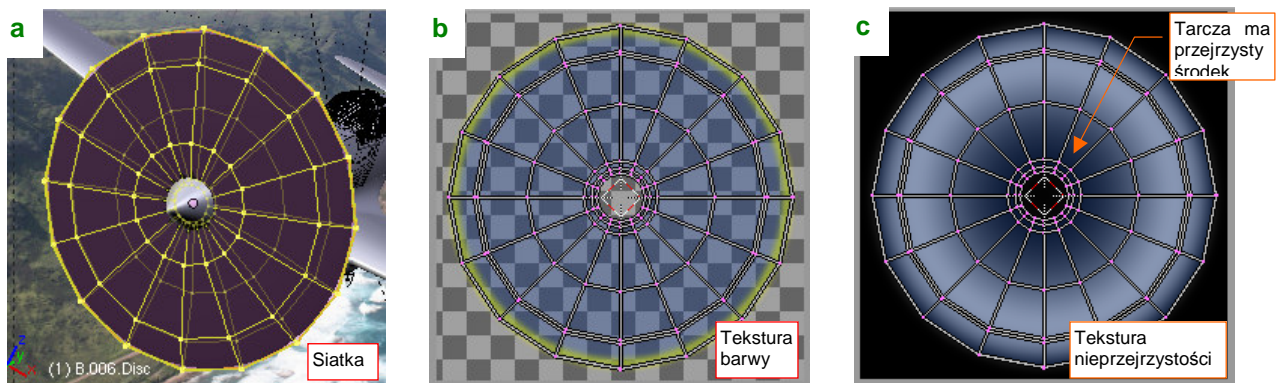
Rysunek 8.1.20 Rezultat złożenia efektów *MBLUR* i *Vector Blur*

Rozmycie jest płynne, bez śladów krawędzi (Rysunek 8.1.20b). *Vector Blur* dobrze sprawdza się w tej roli!

Czy zastanawiałeś się nad tym, dlaczego właściwie staramy się w efekcie rozmycia wirującego śmigła odtworzyć to, co widać na fotografiach? (Wiele razy przewinięły się przez tę sekcję takie pojęcia jak przesłona czy czas ekspozycji). Większości z nas, gdy patrzymy na scenę w locie, takie rozmycie wydaje się naturalne. Dlaczego? Bo znamy samoloty i śmigłowce głównie z fotografii<sup>1</sup>!

Niestety, „fotograficzna” konwencja na obrazach samolotów zadomowiła się w naszych umysłach tak dalece, że realistyczna, jednolicie przezroczysta tarcza śmigła wygląda trochę ...dziwnie. Myślę, że takie „tarcze” wirników mogą jeszcze znaleźć zastosowanie w animacji. (Uzyskanie tego efektu wymaga krótszego czasu renderowania niż rozmycie ruchu). Tak czy siak, pokrótce omówię tutaj, jak można odwzorować taki „krąg śmigła”.

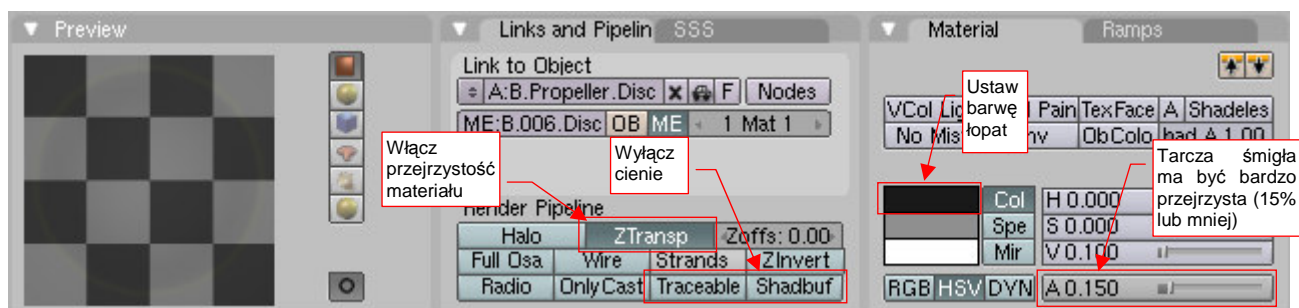
Umieściłem w naszej scenie dysk, którego kontury obejmują tylne i przednie krawędzie łopat śmigła (Rysunek 8.1.21a). Dysk ma więc pewną grubość — taką, jak szerokość łopat w rzucie z boku. Warstwa 17, na której go umieściłem, była do tej pory pusta. Pozwoli nam to włączać jej widoczność tylko w scenach, w których będziemy chcieli użyć tego efektu.



Rysunek 8.1.21 Tarcza, imitująca wirujące śmigło i jej tekstury

Materiał, który pokrywa tarczę, ma kolor łopat śmigła (w tym przypadku to czerni). Dodatkowa tekstura barwy (Rysunek 8.1.21b) nakłada na brzeg dysku żółty akcent (od żółtych końcówek). Najważniejsza jest jednak tekstura nieprzejrzystości (Rysunek 8.1.21c). Moduluje wartość komponentu *Alpha* barwy materiału. Jest nakładana na materiał w trybie *Multiple*. Jej białe obszary zachowują ogólną nieprzejrzystość tarczy, a czarne stają się zupełnie przezroczyste. Zwróć uwagę, że taki właśnie czarny obszar o rozmytych granicach umieściłem w centrum dysku. Maskuję w ten sposób krawędź połączenia tarczy z kołpakiem śmigła.

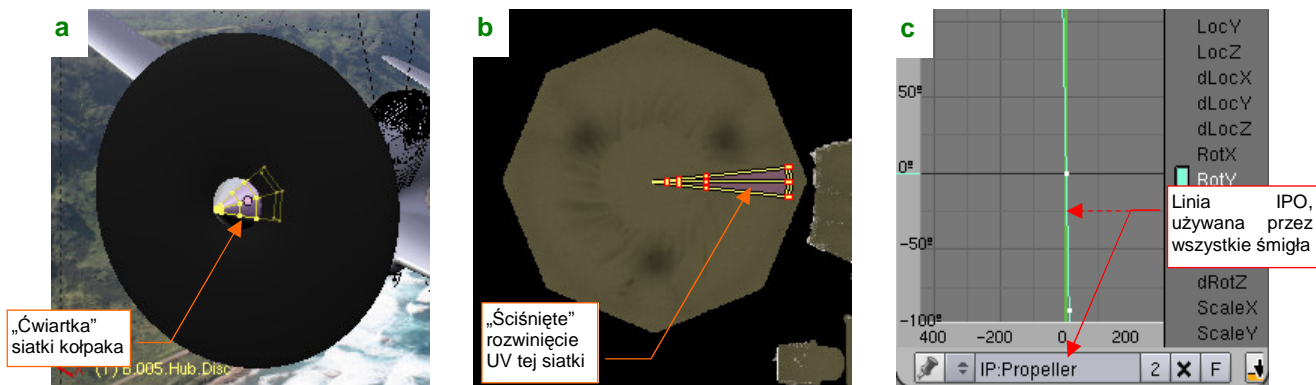
W ustawieniach materiału tarczy, w panelu *Links and Pipeline*, wyłączyłem generowanie cieni (opcje *Shadbuf* i *Traceable*). Włączyłem parametr *ZTransp*. Tarcza będzie miała niewielką przezroczystość: ustawiłem wartość *A(lpha)* na 15% (Rysunek 8.1.22):



Rysunek 8.1.22 Ustawienia materiału tarczy śmigła

<sup>1</sup> Tak się składa, że w ciągu ostatnich dziesięciu lat zdarzało mi się wiele razy lecieć turbośmigłowym ATR-72. Łopaty śmigieł tych samolotów miały takie same barwy, jak łopaty samolotów z lat 40-tych. (Pomalowano je na czarno, z żółtymi końcówkami). ATR-72 także lata na podobnych pułapach i z podobnymi prędkościami, co myśliwce z tego okresu. Podczas lotu zawsze uważnie przyglądałem się śmigłom tej maszyny. Oglądałem je przy różnej pogodzie i oświetleniu. Mogę powiedzieć z ręką na sercu: gdy łopaty wirują, to tworzą jednolite, słabo widoczne koło. Nic więcej! (Taka leciutka, ciemna „mgiełka”). Żółte końcówki są nieco bardziej widoczne, ale nie rzucają specjalnie w oczy.

Oprócz tarczy, na warstwie 17 umieściłem także „wirującą” wersję kołpaka śmigła. Musi mieć jak najbardziej jednolite barwy wzdłuż kierunku obrotu. Dlatego z oryginalnej siatki pozostawiłem tylko „ćwiartkę”, powieloną modyfikatorami *Mirror* w cały stożek (Rysunek 8.1.23a):



Rysunek 8.1.23 Kołpak tarczy śmigła — siatka i animacja

Na rozwinięciu UV dodatkowo „ścisnąłem” ściany tej siatki w bardzo wąski trójkąt (Rysunek 8.1.23b). (Dzięki temu wszystkie zabrudzenia, namalowane na kołpaku, rozciągną się wzdłuż kierunku obrotu). Do kołpaka przypisałem (relacją *Parent*) tarczę śmigła. Sam kołpak „uruchomiłem” krzywą IPO **Propeller**, używaną przez pozostałe śmigła tej sceny (Rysunek 8.1.23c). (To z myślą o ewentualnych animacjach — tarcza będzie lepiej wyglądać, gdy będzie się nieznacznie poruszać).

Teraz w ustawieniach renderowania należy wyłączyć użycie węzłów (opcja *Do Composite* — por. str. 513, Rysunek 8.1.10). Powrócimy w ten sposób do „prostego” generowania obrazu na podstawie pojedynczej sceny. Potem pozostaje nacisnąć *Render* i poczekać chwilę na rezultat (Rysunek 8.1.24):



Rysunek 8.1.24 Efekt z dyskiem, imitującym wirujące śmigło

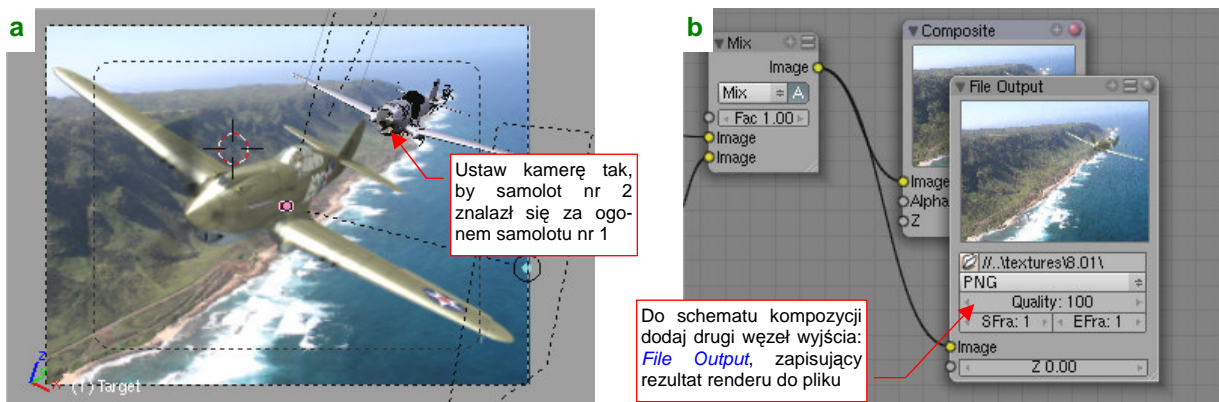
Hmm..., no cóż z tym okrągłym „kółkiem” model wygląda trochę dziecinnie. Widzisz, jak głęboko tkwi w nas świat „widziany fotografiami”? Sądzę, że można jeszcze trochę zmniejszyć przejrzystość tarczy (parametr  $A(\alpha)$ , por. Rysunek 8.1.22) — do, powiedzmy, 10%. Wierz mi czy nie, ale ten samolot tak by wyglądał, gdybyś miał okazję zobaczyć go na własne oczy w locie!

Czy renderowanie modelu z jednolitą tarczą trwało krócej? Owszem, Blender tworzył obraz, który pokazuje Rysunek 8.1.24, przez 2 min 8s. Tę samą scenę, ale z efektem rozmycia łopat śmigła (Rysunek 8.1.20) generował przez 2 min 44s. Oznacza to, że renderowanie modelu „z tarczą śmigła” zajęło raptem o 23% czasu mniej. Mówiąc szczerze, spodziewałem się większej różnicy. (Wygląda na to, że mamy dobrze opracowany efekt rozmycia). Skrócenie czasu renderowania animacji o 20% raczej nie będzie decydować o wyborze efektu dla odwzorowania śmigła. Przy tak niewielkiej różnicy zadecyduje raczej indywidualny gust autora<sup>1</sup>.

Twoje kompozycje nie muszą się ograniczać do jednego samolotu. Możesz tworzyć także sceny z udziałem wielu maszyn. Nie trzeba ich nawet łączyć w jednym pliku Blendera. Często spotykaną metodą „powielenia” modelu jest zapisanie obrazu samolotu do pliku. Możesz do tego użyć np. dodatkowego węzła rezultatu, typu [File Output](#) (Rysunek 8.1.25b — więcej szczegółów na str. 1036). Taki obraz powinien mieć przejrzyste tło. Potem można wstawić go kilka razy w GIMP, w różnej skali, ponad dowolnym obrazem tła. W ten sposób najlepiej wychodzi tworzenie formacji bombowców (bo wszystkie samoloty lecą w nich w tym samym kierunku). Mankamentem tej metody jest jednak ułomność Blendera: na tak wyeksportowanych obrazach brakuje części odbłasków słońca na szybach! To ten sam efekt, o którym wspominałem na str. 515 (por. Rysunek 8.1.14). Refleksy świetlne, widoczne na normalnym tle w [Render Window](#), znikają wszędzie tam, gdzie za szybą jest przezroczyste tło, a nie jakiś fragment samolotu. Potem jest z tym kłopot, bo trzeba jakoś „ręcznie” domalować odbłaski w Gimpie. Przy odrobinie wprawy nie jest to jednak aż tak trudne, i wielu grafików stosuje tę metodę.

Jeżeli jednak chcesz zrobić scenę z parą lub kluczem samolotów — można to zrobić w samym Blenderze, bez użycia Gimpa. Zaczynasz od przygotowania podstawowej sceny z jednym samolotem. Zazwyczaj jest to maszyna, która ma się znajdować najbliżej obserwatora. Dopasowujesz na niej wszystkie szczegóły: oświetlenie, położenie na tle fotografii. Gdy jesteś z całości zadowolony, zachowaj jej ostatni render (przyda się za chwilę). Następnie skopiuj tak przygotowany plik Blendera tyle razy, ile samolotów ma być na ostatecznej scenie. Stworzymy tutaj obraz pary P-40, więc potrzebujemy tylko jednej kopii. Nadaj temu plikowi nazwę, powiedzmy, [2.blend](#) (bo będzie to obraz samolotu „prowadzonego” — nr 2 pary bojowej). Nazwę oryginalnego pliku zmień na [1.blend](#) (bo jest to obraz samolotu „prowadzącego” — nr 1 pary bojowej). Oczywiście, możesz je także nazwać inaczej, ale w każdym razie takich nazw będę używał w dalszym opisie.

W pliku [2.blend](#) ustaw jako tło kamery (nie renderu!) wygenerowany przed chwilą obraz samego samolotu prowadzącego (Rysunek 8.1.25a):



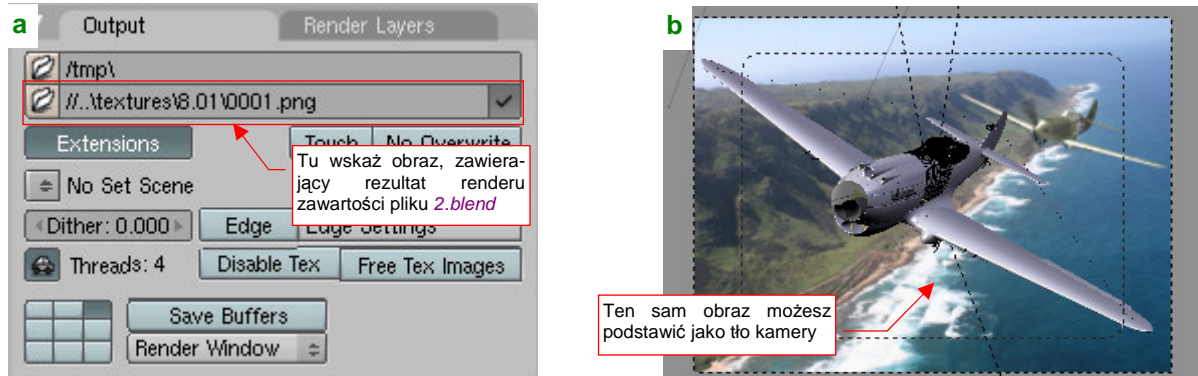
Rysunek 8.1.25 Kompozycja pary samolotów: przygotowanie samolotu dalszego (plik [2.blend](#))

Taka pomoc bardzo ułatwia ustawienie model samolotu prowadzonego. Umieść model (tzn. kamerę) w odpowiednim miejscu sceny. Nasz „nr 2” ma się znaleźć za ogonem samolotu nr 1. Dodaj także do schematu węzeł [Output File](#), zapisujący rezultat renderu sceny z samolotem nr 2 do pliku (Rysunek 8.1.25b).

<sup>1</sup> Spotkałem się też z jeszcze innymi metodami odwzorowania śmigła, opartymi o podobną przejrzystą tarczę. Można, na przykład, namalować „fotograficznie” rozmyte łopaty na jej teksturze barw i przejrzystości. Takie podejście pozwala dodać do obrazu nieco „malarzkiej ekspresji”, niedostępnej w metodach opartych o matematycznie wyznaczone rozmycie ruchu!

**File Output** tworzy we wskazanym folderze plik o nazwie odpowiadającej numerowi aktualnie wybranej klatki (nieco więcej na ten temat — str. 1036). Ustaw w pliku *2.blend* klatkę nr 1, aby podczas renderu powstawał plik o nazwie *0001.png*.

Otwórz teraz plik z samolotem prowadzącym (*1.blend*). Ustaw w nim obraz *0001.png* jako tło renderu (w panelu **Output** — Rysunek 8.1.26a):



Rysunek 8.1.26 Kompozycja pary samolotów: przygotowanie samolotu bliższego (plik *1.blend*)

Obraz tła możesz także podstawić pod widok z kamery (Rysunek 8.1.26b). Zawsze się przecież może zdarzyć, że będziesz chciał trochę poprawić położenie samolotu wiodącego. Wtedy takie tło będzie bardzo przydatne! Pozostaje wykonać render pliku *1.blend*, i mamy rezultat (Rysunek 8.1.27):



Rysunek 8.1.27 Para P-40 z 47 PS w locie nad półwyspem Kaena (Hawaje, wyspa Oahu, tło: [www.hawaii.gov](http://www.hawaii.gov))

Obiekty odleglejsze od obserwatora wydają się jaśniejsze od bliższych. Dlatego, w ramach „drobnej kosmetyki”, możesz rozjaśnić barwę cieni samolotu prowadzonego (*2.blend*) o jakieś 10%. (Chodzi o kolor **Ambient** — por. str. 510, Rysunek 8.1.5). Nie zapomnij też przygotować dla samolotu nr 2 indywidualnej tekstury *decals.png*, z innym numerem taktycznym! Mimo, że obserwator widzi burty samolotów pod bardzo małym kątem, to jednak są nadal czytelne!



## Podsumownie

- W tej sekcji pokazałem, jak stworzyć scenę w locie poprzez wkomponowanie modelu w fotografię. Zdjęcia, których do tego używam, znajdują w Internecie. Korzystam z wyszukiwarek, by znaleźć darmowe fotografie o odpowiedniej rozdzielczości. Na przykład Google pozwala ustawić min. rozmiar obrazu w kryteriach wyszukiwania;
- „Pozowanie” samolotu odbywa się poprzez przesunięcia kamery, albo przesunięcie lub obrót jej „celu” (obiektu **Target**). Dużą pomocą jest ustawienie fotografii tła w oknie widoku z kamery (jako *Background Image* — str. 509);
- Po ustaleniu pozycji i orientacji kamery, trzeba ustalić pozycję i energię światła. Chodzi o odtworzenie warunków oświetlenia widocznych na fotografii (str. 509). Oprócz dwóch światła podstawowych — *Sun* i *Hemi* — możesz wykorzystać słabe światło pomocnicze (także typu *Hemi* — str. 510). Za pomocą takiego światła można uzyskać na powierzchni modelu dodatkowe odbłaski i rozjaśnienia. Często przydaje się także zmienić trochę barwę otoczenia (*Ambient* — str. 510) (Można ją także określić jako „kolor cieni”);
- Efekt „rozmycia ruchu” łopat śmigła jest kanonem, który powstał i utrwalił się dzięki fotografii (str. 519). Nie sposób się od niego uwolnić. Aby go uzyskać, musisz najpierw animować obrót zespołu śmigła (str. 511). Typowa „prędkość” tego ruchu to 50-65 klatek na pełen obrót (str. 511 — 512). (Wartość dobrana do efektu *MBLUR* — *Motion Blur*).
- Efekt *MBLUR* (*Motion Blur*) tworzy poprawny obraz rozmycia ruchu, ale wiąże się z koniecznością renderowania całej sceny 16 razy (str. 512). Uważam, że zabiera to za dużo czasu. Obejściem problemu jest podział modelu na dwie sceny. W jednej renderowane jest tylko element wirujący — samo śmigło. Ta scena ma włączone rozmycie ruchu (*MBLUR*). Druga scena to cała reszta samolotu, renderowana bez *MBLUR* (str. 512, 514). Potem obydwa obrazy są łączone za pomocą odpowiedniego schematu kompozycji (*Composite Nodes*). Taki podział pozwala skrócić czas uzyskania efektu 10 razy (z 14 minut do 1 min 40s). Alternatywny efekt — *Vector Blur* — jest za słaby. Nadaje się wyłącznie do dodatkowego rozmycia obrazu śmigła, uzyskanego za pomocą *MBLUR* (str. 514, 518);
- Przy okazji prób skrócenia czasu renderowania rozmycia ruchu poznaliśmy schematy węzłów Blendera (str. 513, 514, 516, 518). Do pracy z nimi służy okno *Node Editor*. To bardzo elastyczne narzędzie o dużych możliwościach. Pozwala na łączenie ze sobą wielu różnych obrazów, poddając je przedtem różnorodnym transformacjom;
- Typowym węzłem, dostarczającym schematom kompozycji Blendera danych wejściowych jest *Render Layer*. Reprezentuje na schemacie pojedynczą „warstwę renderowania” (*Render Layer*). Każda scena Blendera może mieć wiele różnych warstw renderowania. Zresztą nazwa jest trochę myląca — może bardziej poprawną byłby może „przebieg renderera” (*render pass*). Możesz zdefiniować wiele alternatywnych warstw renderowania. Każda z nich to indywidualny zestaw parametrów dla renderera. Opisuje, co ma być przetworzone, oraz wskazać rodzaje danych udostępnianych na wyjściu (str. 513). Domyślnie każda scena Blendera ma jedną taką warstwę, o nazwie **1 Render Layer**;
- Wirujące śmigło wygląda w ludzkich oczach (a nie na fotografii) jak jednolity, lekko nieprzejrzysty dysk. Na str. 519 — 520 pokazuję, jak osiągnąć ten efekt. Może się czasami przydać — np. w sekwencji filmowej;
- Istnieje wiele metod stworzenia sceny z kilkoma samolotami. Na str. 521 — 522 pokazuję, jak uzyskać obraz pary P-40. Używam do tego dwóch plików Blendera (po jednym na każdy samolot);

## 8.2 Realistyczna scena na ziemi

Choć może wydawać się to dziwne, trudno jest znaleźć duże, dobre zdjęcie lotniska polowego, które nadaje się na tło sceny. Szczególnie, jeżeli szukamy fotografii z określonej okolicy, z której operował nasz model. W tej sekcji „postawimy” na lotnisku polowym P-40 („białą 75”) z AVG. Szukałem przez parę dni dużych zdjęć z prowincji Yunnan (Chiny), ale niczego odpowiedniego nie znalazłem. Jedyne, co mogłem po takiej „foto-wycieczce” stwierdzić, to, że lasy i zarośla tych górskich obszarów nie różnią się specjalnie od drzew i krzewów znanych nam ze strefy umiarkowanej. Szczególnie, jeżeli patrzymy na nie z większej odległości, np. ze środka jakiegoś dużego pola. Kierując się tym spostrzeżeniem, wybrałem na tło naszego samolotu następującą fotografię (Rysunek 8.2.1):

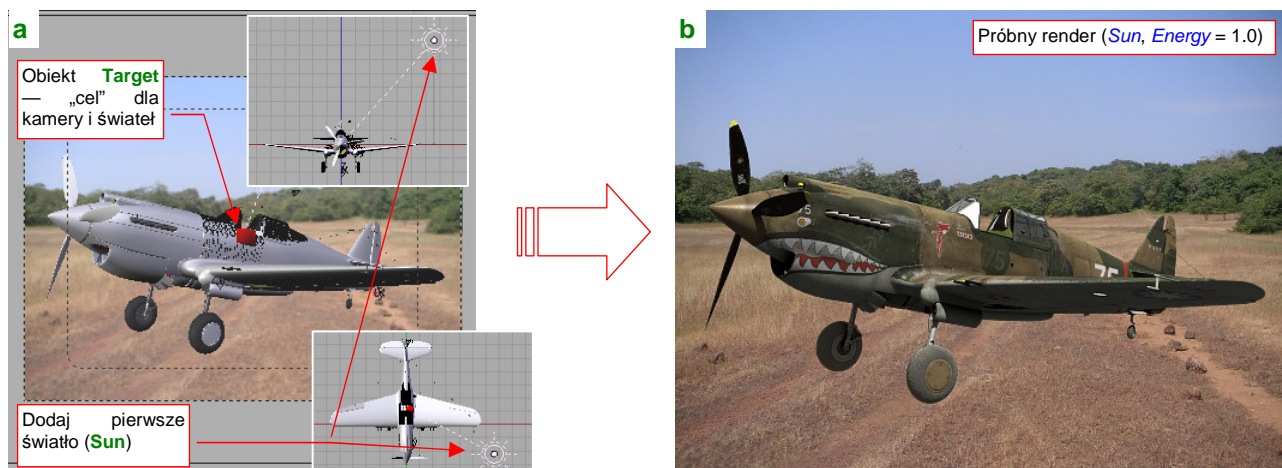


Rysunek 8.2.1 Nasz zakątek lotniska (zdjęcie z [blogs.bootsnall.com](https://blogs.bootsnall.com))

Obawiam się, co prawda, że to zdjęcie przedstawia kawałek ziemi z zupełnie innego regionu świata (może Afryki Południowej?). Założmy jednak, że jest to ustronny zakątek lotniska w Kunming lub jakiegoś tymczasowego lądowiska, z którego także operowały pary samolotów dywizjonu „Hell’s Angeles”. Oryginalną fotografię poddałem najpierw małej „obróbce” w GIMP. Obraz został przerzucony w poziomie, oraz uległ wyostrzeniu (*Filters* → *Enhance* → *Sharpen*) o 40%. (Takie wyostzone krawędzie lepiej pasują do linii renderu, wygładzanych „twardym” filtrem *CatRom* — por. str. 971) .

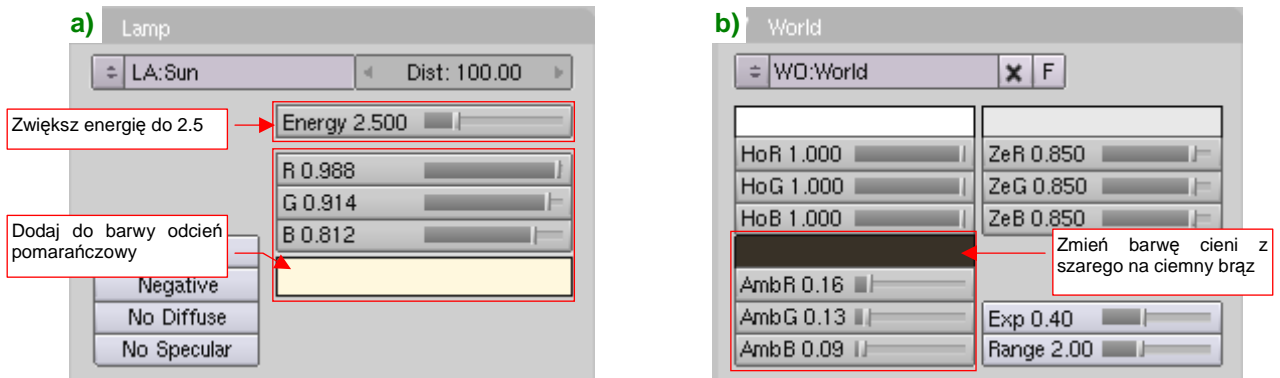
Podstaw to zdjęcie jako tło sceny. Kamerę ustaw tak, by koła modelu „stanęły” na gruncie (Rysunek 8.2.2). (To wymaga pochylecia „celu” kamery — włącz w jej ograniczeniu *Track To* przełącznik *TargetZ*).

W poprzedniej sekcji, omawiającej scenę w locie, „prześlizgnąłem” się pośpiesznie ponad aranżacją świateł. (Ustawialiśmy je już w poprzednich rozdziałach — por. str. 285, 271). W scenie na ziemi trzeba „poustawiać” je inaczej. Pokażę więc teraz budowanie nowego systemu oświetlenia od podstaw, krok po kroku. Zaczniemy od dodania słońca (światła typu *Sun*). W tym przypadku umieść je ponad kamerą (Rysunek 8.2.2a):



Rysunek 8.2.2 Ustawienie kamery i słońca

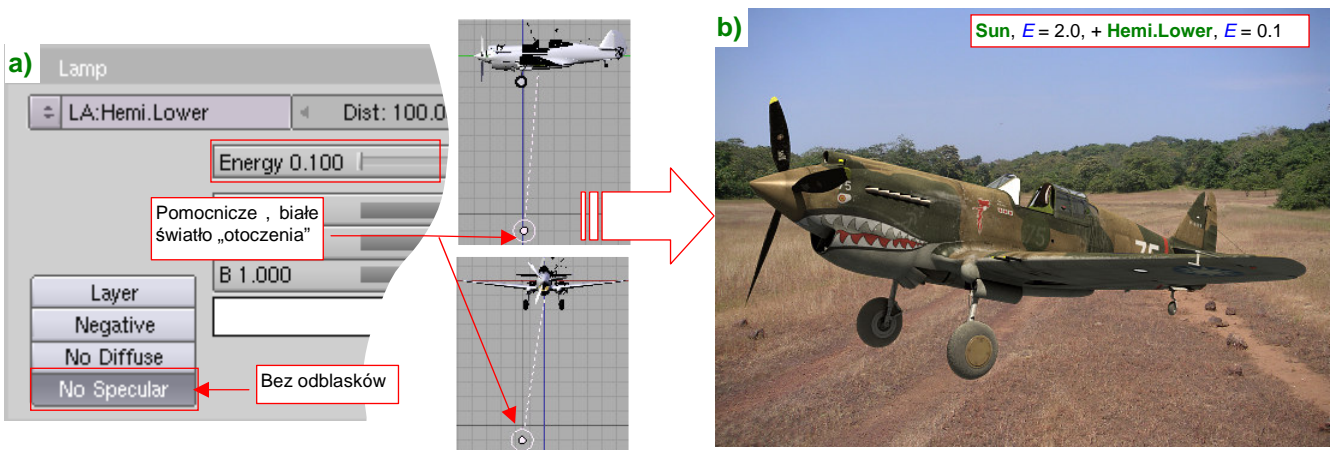
Samolot na próbnym renderze (Rysunek 8.2.2b) jest zbyt ciemny. Aby to skorygować, zwiększ energię światła **Sun** do 2.5. Przy okazji, aby dopasować „temperaturę” oświetlenia do zdjęcia, zmień także jego barwę, z białej na „bardzo wyblakły pomarańczowy” (Rysunek 8.2.3a):



Rysunek 8.2.3 Poprawki w barwie i sile oświetlenia

Z tego samego powodu dodaj do barwy światła otaczającego (cieni) nieco żółci i czerwieni (Rysunek 8.2.3b).

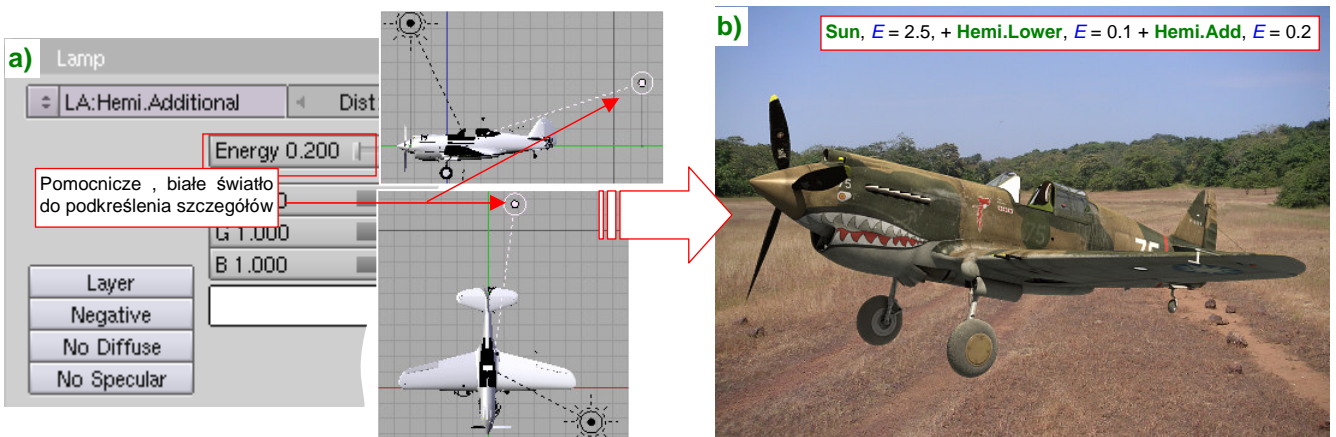
Teraz dodamy słabe światło typu **Hemi** ( $Energy = 0.1$ ), oświetlające model od dołu (Rysunek 8.2.4a). Nazwałem je **Hemi.Lower**. Służy to rozproszeniu zbyt gęstych cieni na spodzie samolotu (Rysunek 8.2.4b):



Rysunek 8.2.4 Podświetlenie spodu samolotu

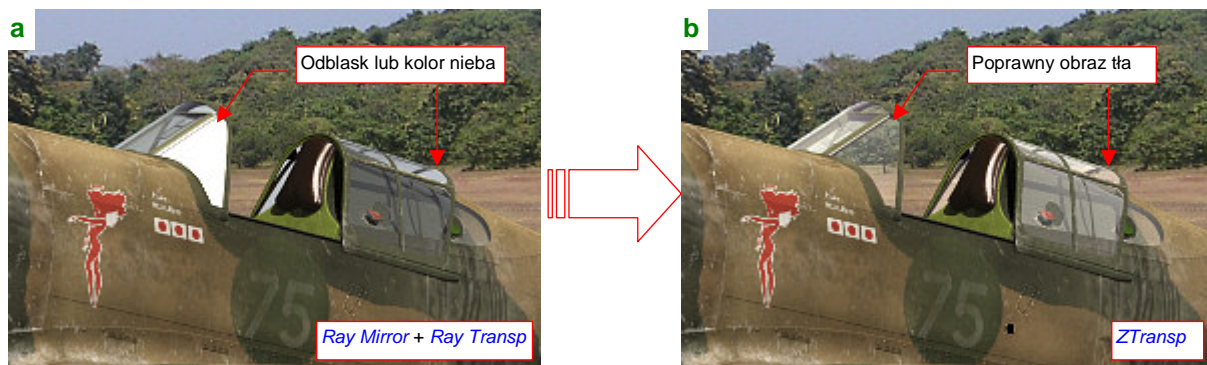
Aby na spodzie modelu nie pojawiły się odbłyski, wyłącz je (przełącznikiem **No Specular** — Rysunek 8.2.4a).

Ostatnią „lampą” na tej scenie jest druga, pomocnicza hemisfera o nazwie **Hemi.Additional** (Rysunek 8.2.5a). To typowe, słabe ( $Energy = 0.2$ ) białe światło „do podkreślenia walorów powierzchni”. Ustawiłem je tak, by swoimi odbłaskami rozjaśniło piastę i goleń prawego koła, oraz uwydatniło zagłębienia na poszyciu steru kierunku (Rysunek 8.2.5b):



Rysunek 8.2.5 Podświetlenie podwozia i steru kierunku

Światła na scenie mamy już ustawione, czas wziąć się za kolejny szczegół: oszklenie kabiny. Widoczny na niej odbłask, rozciągnięty na cały bok wiatrochronu, wydaje się przesadzony (Rysunek 8.2.6a):



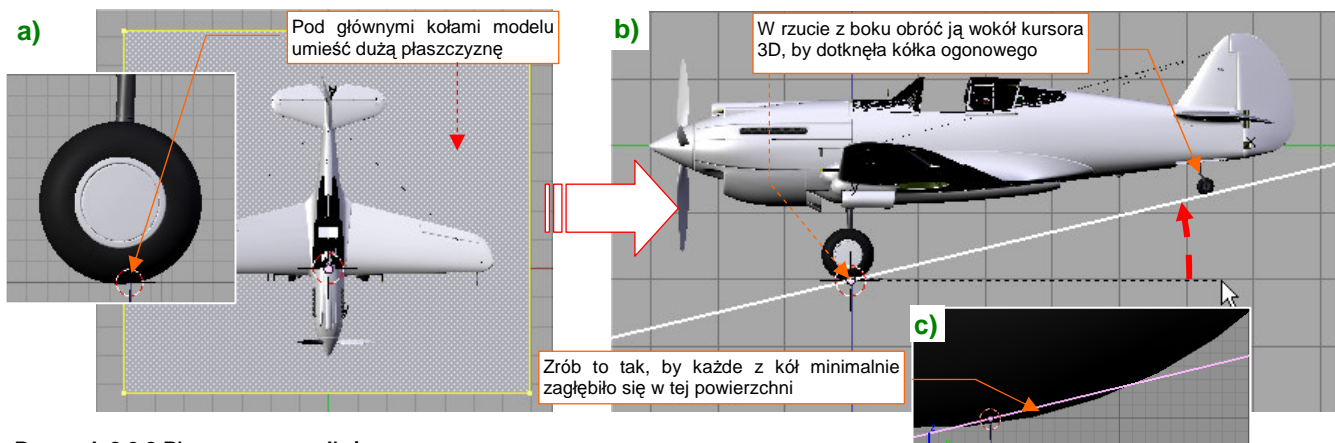
Rysunek 8.2.6 Zmiana materiału „szkła” na uproszczony model (**ZTransp**)

Co gorsza, to nie jest kwestia tylko zmniejszenia połyskliwości. Popatrz w inne miejsca owiewki, przez które powinny prześwitywać drzewa lub trawa lotniska. Nic takiego tam nie ma! Zamiast tego widać tam... niebo. To błąd Blendera, o którym już wspominaliśmy przy okazji konfiguracji materiału na oszklenie kabiny (por str. 981, 982). W tym przypadku mniejszym złem będzie zastosowanie tej samej recepty, którą podałem w tamtej sekcji. Wyłącz w materiale **B.Glass.Canopy** opcje **Ray Transp** i **Ray Mirror** (panel **Mirror Transp**), i zastąp je uproszczonym modelem przejrzystości (opcją **ZTransp**). Co prawda zupełnie znikną z szyby odbłaski słońca (Rysunek 8.2.6b), ale i tak sądzę, że w tym ujęciu powinny być minimalne. Po tej korekcie uległ skróceniu czas renderu: z 2 min 33s do 1 min dla **ZTransp**. Szczegóły zmian w materiale przedstawia Rysunek 8.2.7:



Rysunek 8.2.7 Szczegóły zmiany materiału szkła

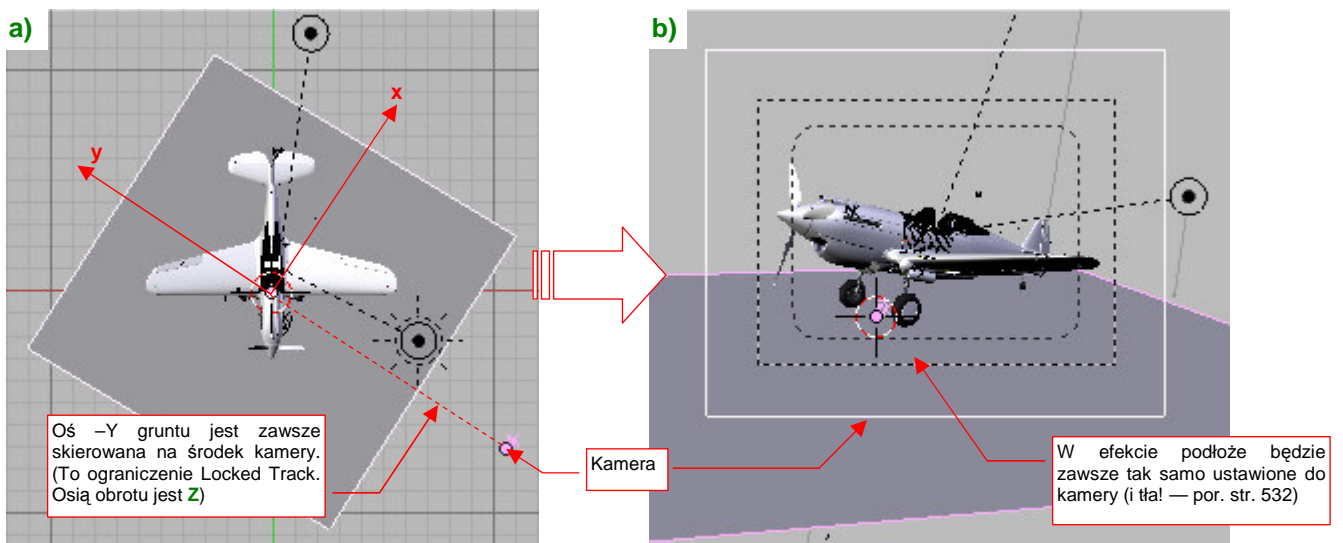
„Wierzch” samolotu już całkiem dobrze pasuje do tła (por. chociażby Rysunek 8.2.6b). Teraz musimy zająć się jego spodem, a dokładniej: stworzyć na zdjęciu cień naszego modelu. Umieść kursor 3D w środku samolotu, ale na poziomie kół (Rysunek 8.2.8a). Wstaw w tym miejscu duży kwadrat (**Plane**) — to będzie nasz „grunt”:



Rysunek 8.2.8 Płaszczyzna podłoża

Nadaj temu obiektowi nazwę **Ground**, i umieść na warstwie 17. Podstaw go, w rzucie z góry, pod koła główne. Potem przełącz się na rzut z boku, i obróć tę płaszczyznę tak, by dotknęła kółka ogonowego (Rysunek 8.2.8b). Upewnij się, że wszystkie koła minimalnie przenikają przez jej powierzchnię (Rysunek 8.2.8c).

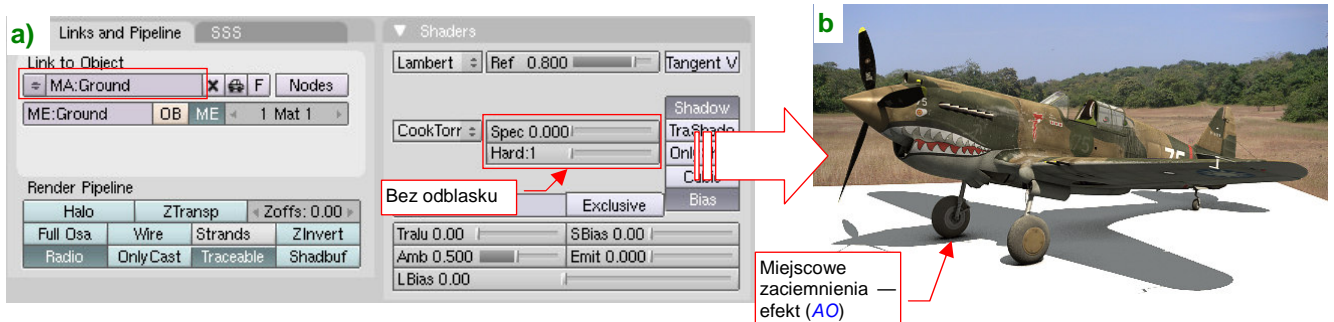
Przy okazji możesz naszemu „gruntowi” przypisać ograniczenie *Locked Track*, kierujące na kamerę oś  $-Y$  tej płaszczyzny (Rysunek 8.2.9a):



Rysunek 8.2.9 Skierowanie podłoża na kamerę (za pomocą ograniczenia *Locked Track*)

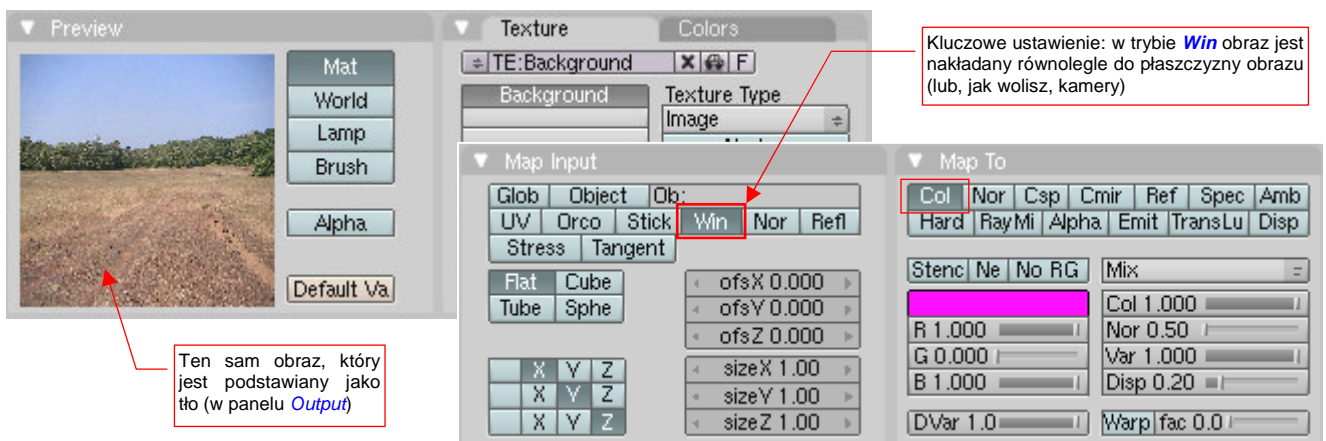
Oś obrotu tego ograniczenia jest lokalna oś  $Z$  podłoża. Takie ustawienie pozwala na obracanie modelu w kamerze (a właściwie — kamery wokół modelu) bez zmiany tła (por. str. 535, Rysunek 8.2.34). To bardzo ułatwia późniejsze poprawki sceny.

Przypiszmy do powierzchni **Ground** materiał (na razie biały). Wyłącz na nim wszelkie blaski (*Shaders:Spec*) i cienie (*Links and Pipeline:Shadbuf*), które może rzucać sama płaszczyzna (Rysunek 8.2.10a):



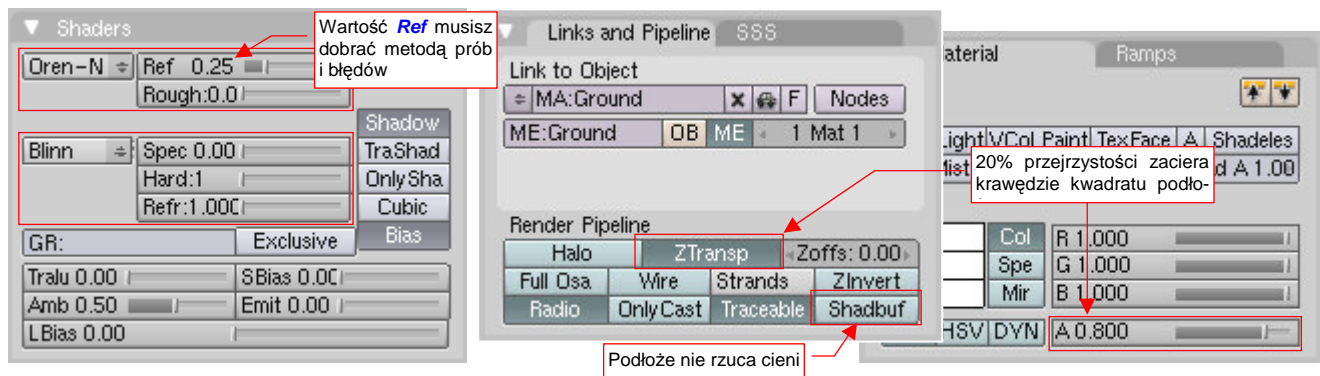
Rysunek 8.2.10 Przymiarka podłoża

Wykonaj próbny render, by sprawdzić, gdzie jak wygląda na niej cień samolotu (Rysunek 8.2.10b). Barwę podłoża nadamy za pomocą tekstury (dlatego materiał **Ground** miał kolor biały — by jej nie przyciemniać). Nałóż na tę powierzchnię obraz tła, stosując specjalny rodzaj mapowania: *Win* (Rysunek 8.2.11):



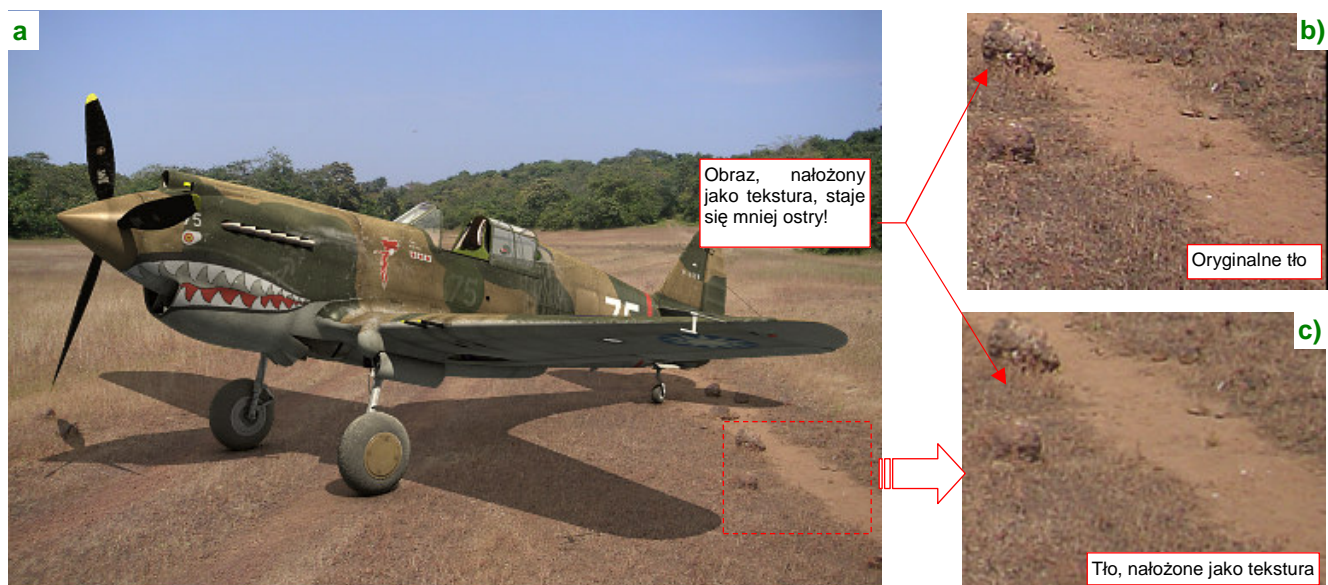
Rysunek 8.2.11 Wykorzystanie obrazu tła jako tekstury

W trybie **Win** obraz tekstury jest „przyczepiony” nie do obiektu, ale do kamery! (To tak, jak gdybyś wyświetlał na siatce modelu obraz z rzutnika slajdów). Przy takim mapowaniu każdy piksel tekstury jest tam, gdzie na obrazie tła. Jeżeli jeszcze dobierzesz (niestety, za pomocą kilku „prób i błędów”) odpowiednią wartość **Ref** materiału (Rysunek 8.2.12), to uzyskasz doskonałe scalenie naszego podłoża z resztą zdjęcia (Rysunek 8.2.13a):



Rysunek 8.2.12 Szczegóły dopasowania materiału podłoża (Ground)

Uzyskaliśmy w ten sposób nasz upragniony cień! No, z tym idealnym scaleniem z obrazem to nie jest tak do końca, jak napisałem. Gdzieś tam na renderze możesz zauważyć cieniutkie na 1 piksel linie, biegnące wzdłuż krawędzi siatki podłoża. Na szczęście łatwo je wyeliminować. Musisz tylko nadać materiałowi **Ground** niewielką przezroczystość — włączając opcję **Links and Pipeline:ZTransp** i ustawiając **Material:A**(lpha) na 80% (por. Rysunek 8.2.12). Aha, i koniecznie opcja **Shadbuf**, z panelu **Links and Pipeline**, musi być wyłączona. Jeżeli o tym zapomnisz — zepsuje cały efekt, który pokazuje Rysunek 8.2.13a):



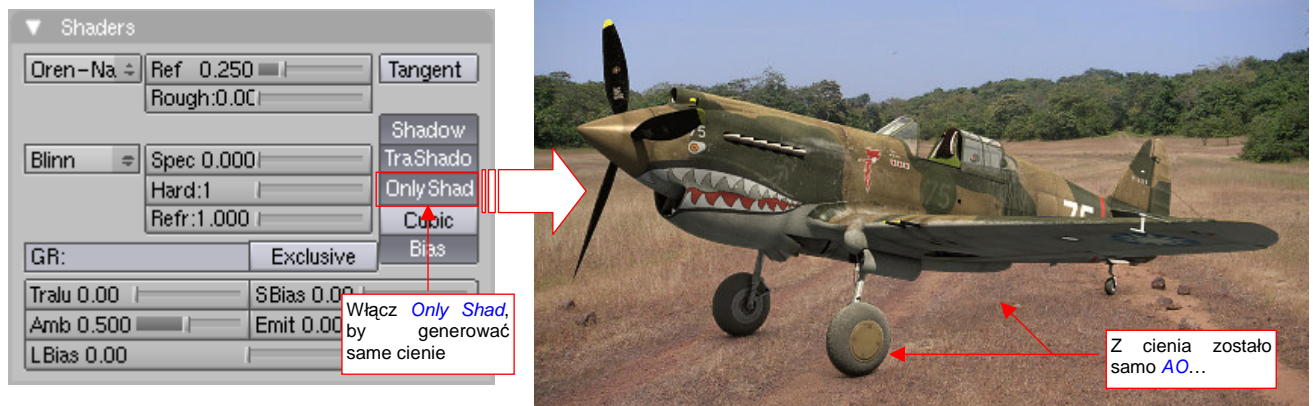
Rysunek 8.2.13 Rezultat mapowania tekstury w trybie **Win**(dow)

Czy na pewno to jest już dokładnie to, o co nam chodziło? Przyjrzyj się uważnie źdźbłom trawy pod samolotem. Nie wyglądają tak ostro, jak na oryginalnym obrazie (por. Rysunek 8.2.13b i c). To rozmycie co prawda zanika, jeżeli zastosujesz teksturę o większej (np. dwukrotnie) rozdzielczości niż tło obrazu. Innym mankamentem jest brak możliwości uzyskania ciemniejszych cieni — bo jak zwiększymy wartość **A**(lpha) materiału, znów pojawią się na renderze cienkie linie wzdłuż krawędzi gruntu<sup>1</sup>. W każdym razie, nie jestem zadowolony z tego efektu.

- Uzyskanie cienia modelu za pomocą powierzchni z obrazem tła nałożonym w trybie **Win** warto zapamiętać jako prostą i szybką metodę. Myślę, że w wielu przypadkach może okazać się wystarczająca.

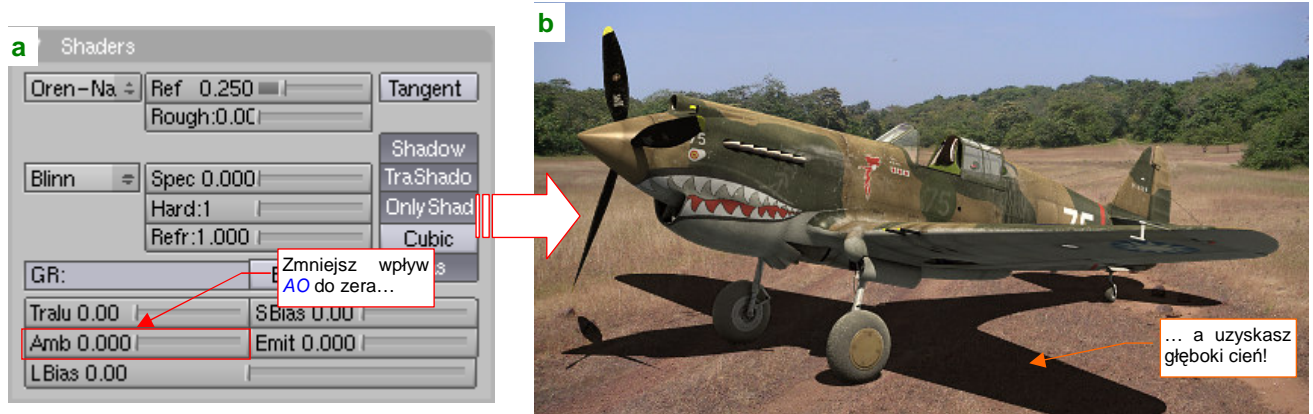
<sup>1</sup> Co prawda, te krawędzie można „rozmyć”, nakładając na powierzchnię kolejną teksturę, modulującą przejrzystość brzegów siatki.

Dokładniejszym, choć w ostatecznym rachunku nieco bardziej pracochłonnym, sposobem jest wykorzystanie opcji **Shaders:Only Shad** (ows). Usuń teksturę z materiału **Ground** i włącz ten efekt (Rysunek 8.2.14a):



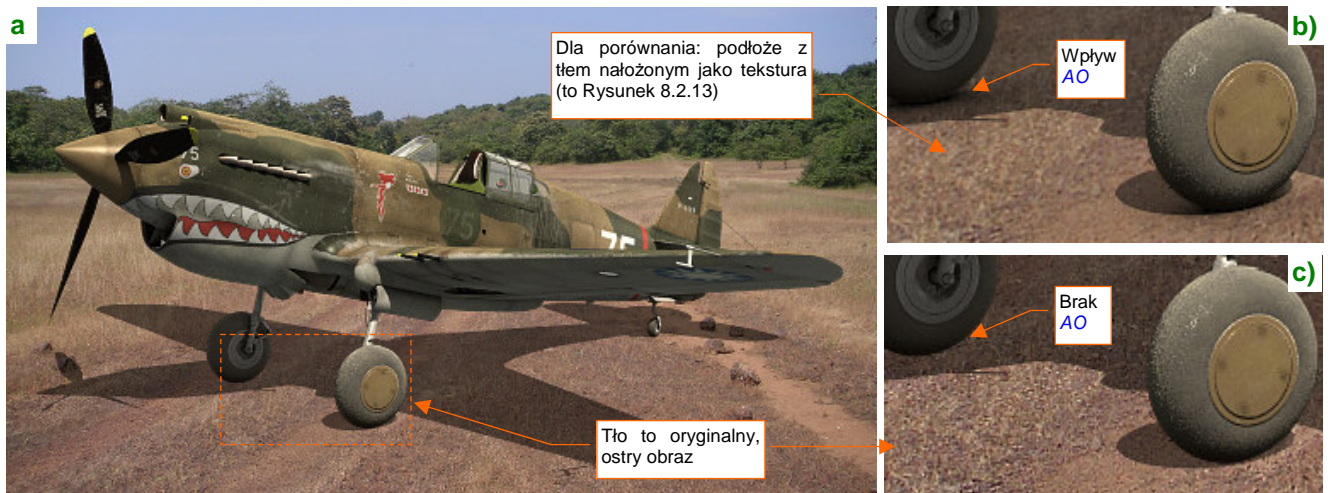
Rysunek 8.2.14 Inna możliwość uzyskania cienia bez tekstury — przełącznik **Only Shad**(ows)

Hmm, w tutorialach pisano, że pod samolotem powinien pojawić się cień, a tutaj widać tylko drobne ślady po **Ambient Occlusion** (Rysunek 8.2.14b). Choć dzięki nim model wydaje się jakoś stać na tym polu (porównaj ze str. 525, Rysunek 8.2.5), to jednak nie o to nam chodziło! Okazuje się, że w ustawieniach materiału trzeba zmniejszyć **Shaders:Amb** dokładnie do zera (Rysunek 8.2.15a):



Rysunek 8.2.15 Rozwiązanie problemu — wyłączenie wpływu **AO** na materiał **Ground**

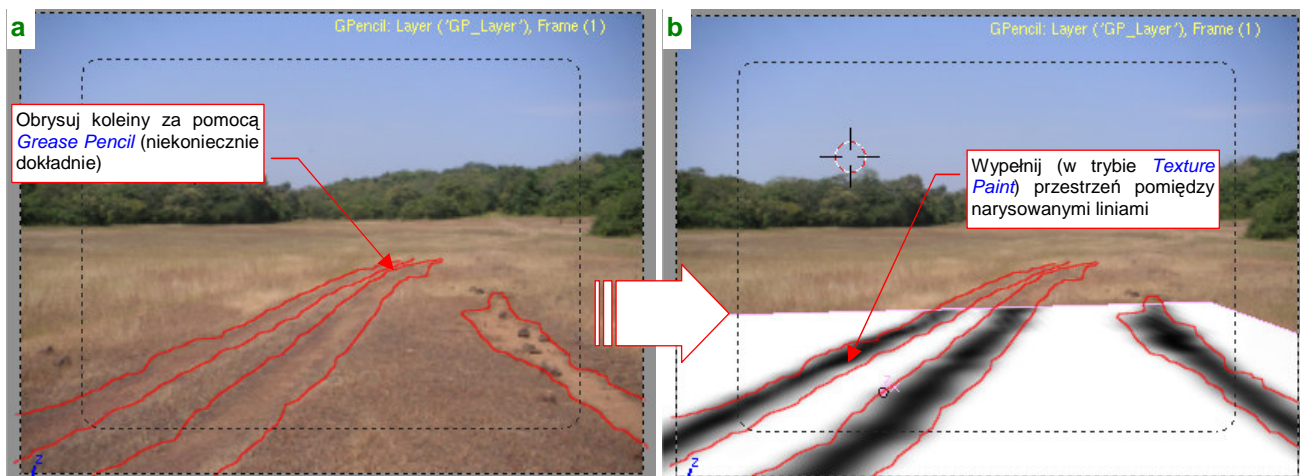
Teraz uzyskaliśmy głęboki, czarny cień (Rysunek 8.2.15b). Możesz go rozjaśnić, zmniejszając wartość **Shaders:A(lpha)** do, powiedzmy, 50% (Rysunek 8.2.16a):



Rysunek 8.2.16 Analiza rezultatu (cień jest nieco rozjaśniony — **Material:A(lpha) = 0.5**)

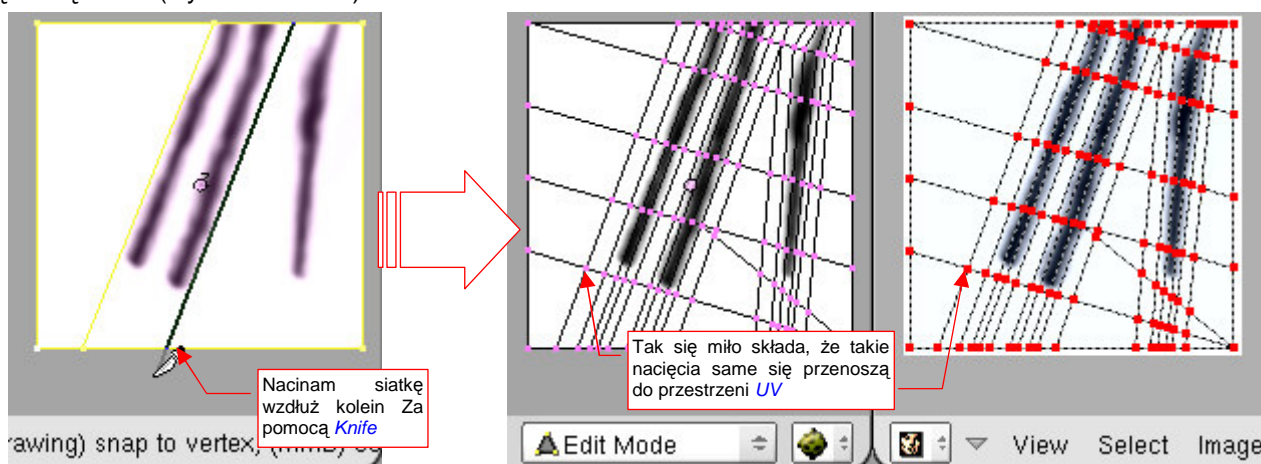
Porównajmy obecny rezultat (Rysunek 8.2.16c) z tym, uzyskanym za pomocą tekstury (Rysunek 8.2.16b). Wygląda na to, że otrzymaliśmy ostry i wyraźny obraz tła. „Zgubiliśmy” jednak pewien drobiazg, dodający realizmu: efekt **Ambient Occlusion**. O tym, jak go przywrócić, napiszę za chwilę (na str. 531).

Na razie wykonamy inny „krok w stronę realizmu”: przez obraz bieżą koleiny. Cień samolotu, który na nie pada, powinien być odpowiednio zniekształcony. Powyginamy w takim razie siatkę podłoża modyfikatorem *Displace* (por. 925), by osiągnąć ten efekt. Niezbędny obraz tekstury narysujemy od razu, w Blenderze (Rysunek 8.2.17):



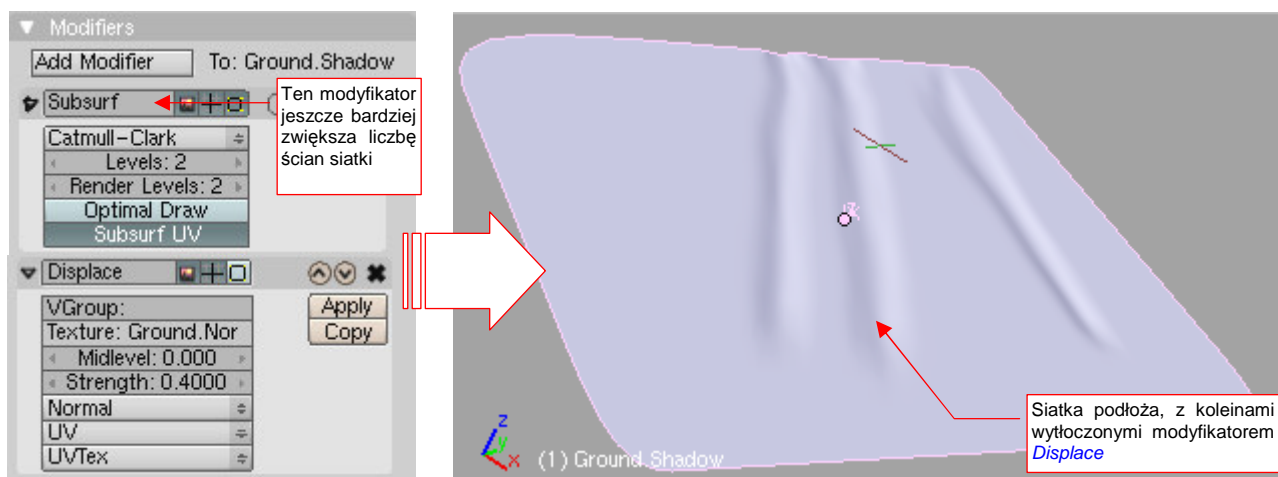
Rysunek 8.2.17 Przeniesienie zarysu kolein na obraz tekstury podłoża

Koleiny, widoczne na obrazie tła, obrysuj za pomocą *Grease Pencil* (Rysunek 8.2.17a). Te linie pomogą Ci namalować obszary wgłębień na obrazie, nałożonym na siatkę podłoża (Rysunek 8.2.17b). (Więcej na ten temat — str. 1037 i następne). Potem podziel siatkę tak, by modyfikator *Displace* miał do dyspozycji odpowiednio dużą liczbę ścian (Rysunek 8.2.17):



Rysunek 8.2.18 Zagęszczanie siatki podłoża w okolicach wgłębień

By jeszcze bardziej zwiększyć ilość ścian, dodałem przed *Displace* modyfikator *Subsurf* (Rysunek 8.2.19):



Rysunek 8.2.19 Naniesienie kolein na siatkę za pomocą modyfikatorów

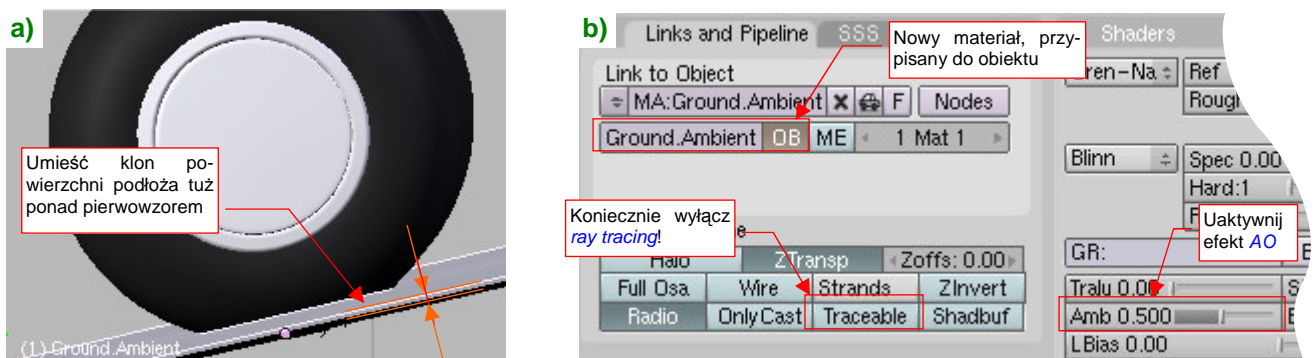


Rezultat jest bardzo dyskretny (Rysunek 8.2.20) — ale realizm składa się także z takich małych szczegółów:



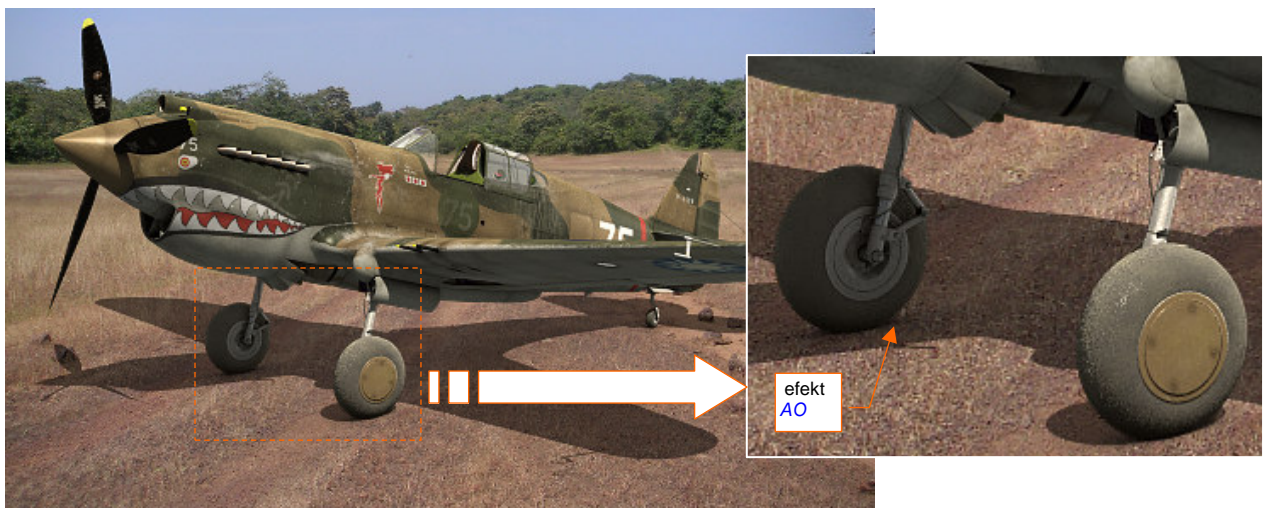
Rysunek 8.2.20 Podłoże z zagłębieniami

Teraz, gdy siatka podłoża ma odpowiedni kształt, stwórz jej klon, i umieść o centymetr (0.1 jedn. Blendera) wyżej (Rysunek 8.2.21a). To będzie dodatkowa powierzchnia, za pomocą której dodamy brakujący efekt **AO**:



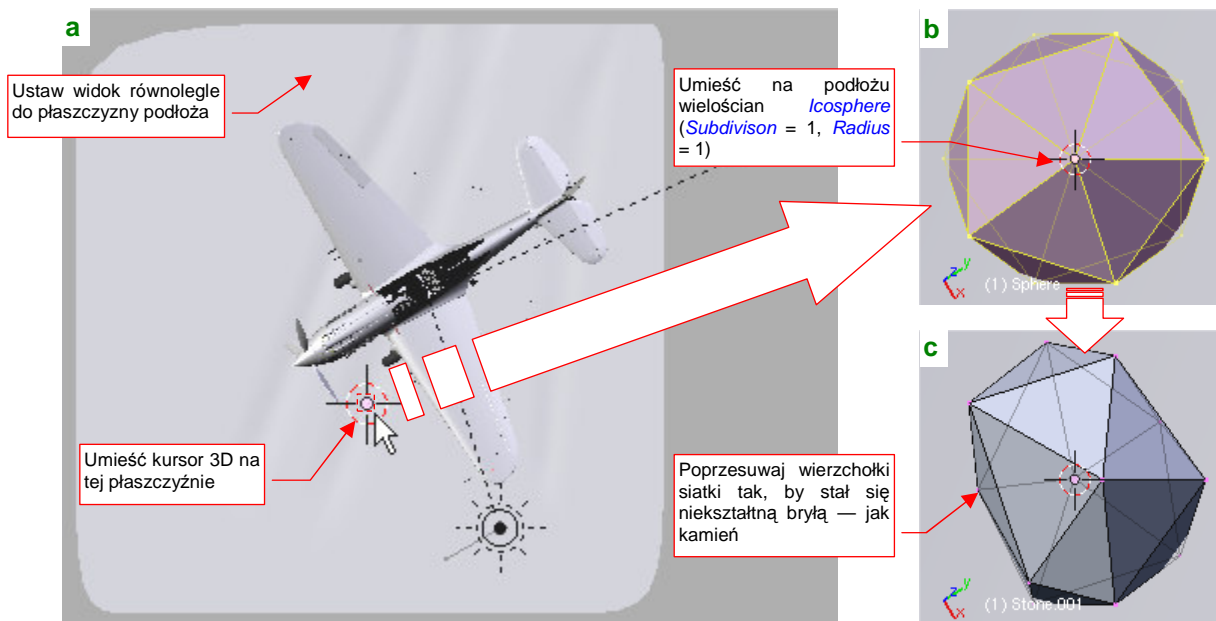
Rysunek 8.2.21 Drugie podłoże — dla „zagęszczenia” cieni przy kołach (efekt **Ambient Occlusion**)

Aby rozróżnić obydwa podłoża, zmień nazwę oryginalnego na **Ground.Shadow**. Nowe nazwij **Ground.Ambient** i umieść na osobnej warstwie (18). Zmień także nazwę materiału **Ground** na **Ground.Shadow**. Przypisz jego kopię do obiektu (nie siatki, bo ta jest wspólna!) **Ground.Ambient**. Nazwij ją **Ground.Ambient** i włącz na niej efekt **AO** (*Shaders:Amb* = 0.5) . Koniecznie wyłącz także opcję *Links and Pipeline:Traceable* (Rysunek 8.2.21b), bo inaczej zepsuje cały efekt, który pokazuje Rysunek 8.2.14:



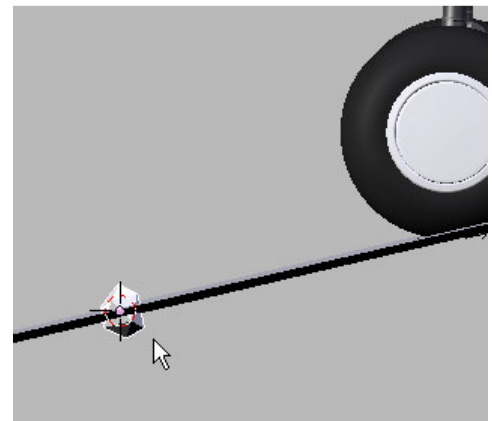
Rysunek 8.2.22 Złożenie obydwu efektów (cień + **AO**)

Przyjrzyj się teraz, czy przypadkiem na fotografii tła nie ma jakichś kamieni, które mogą zmienić granice cienia. W tym przykładzie jest takich kilka — pod lewym statecznikiem i w płaszczyźnie śmigła. Ustaw aktualny widok równoległe do powierzchni podłoża (**Ground.Shadow**), i umieść na nim kursor 3D ( Rysunek 8.2.23a). Następnie wstaw „wzorcowy kamień”: wielościan zbudowaną z trójkątów (**Icosphere** — Rysunek 8.2.23b):



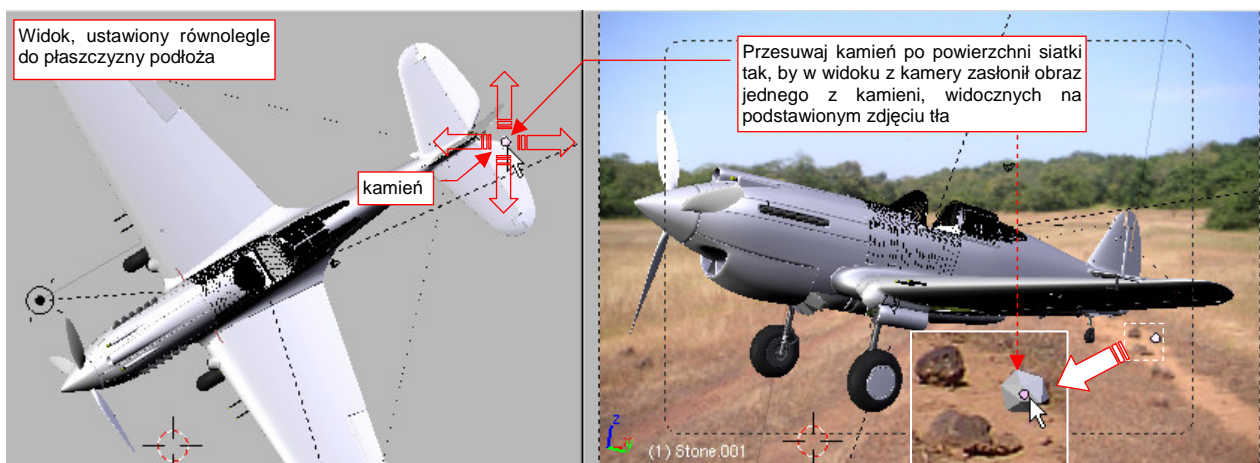
Rysunek 8.2.23 Wstawienie „wzorcowego” kamienia

Poprzesuвай losowo wierzchołki tej siatki, by zaczęła wyglądać jak nieforemny kawałek skały ( Rysunek 8.2.23c). Na koniec upewnij się, że obiekt jest odpowiednio „zagłębiony” w podłożu (Rysunek 8.2.24). Nadaj siatce nazwę **Stone**, a samemu obiektowi — **Stone.001** (bo będzie ich kilka). Umieść go na tej samej warstwie, co podłożo (17).



Rysunek 8.2.24 Kamień na płaszczyźnie „gruntu”

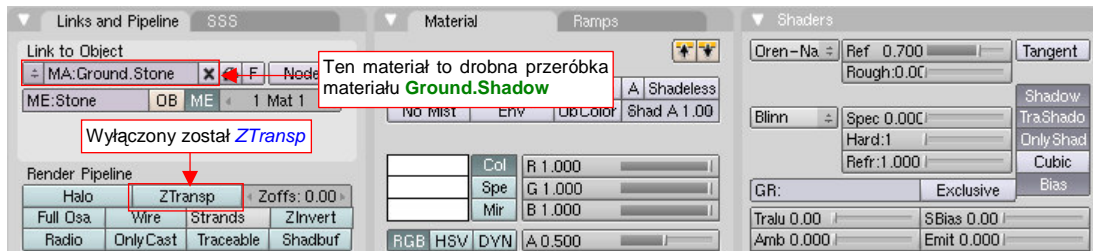
Teraz trzeba ustawić ten obiekt tak, by zasłonił jeden z kamieni widocznych na fotografii. Umieść obok siebie dwa okna **3D View** (Rysunek 8.2.25). W lewym zachowaj projekcję równoległą do podłoża. W prawym ustaw widok z kamery, z włączonym obrazem tła. Następnie zaznacz i ukryj podłożo (H). Potem przesuвай w lewym oknie kamień tak, by w prawym zakrył widoczny na fotografii pierwowzór (Rysunek 8.2.25):



Rysunek 8.2.25 Podstawianie kamieni pod zdjęcie

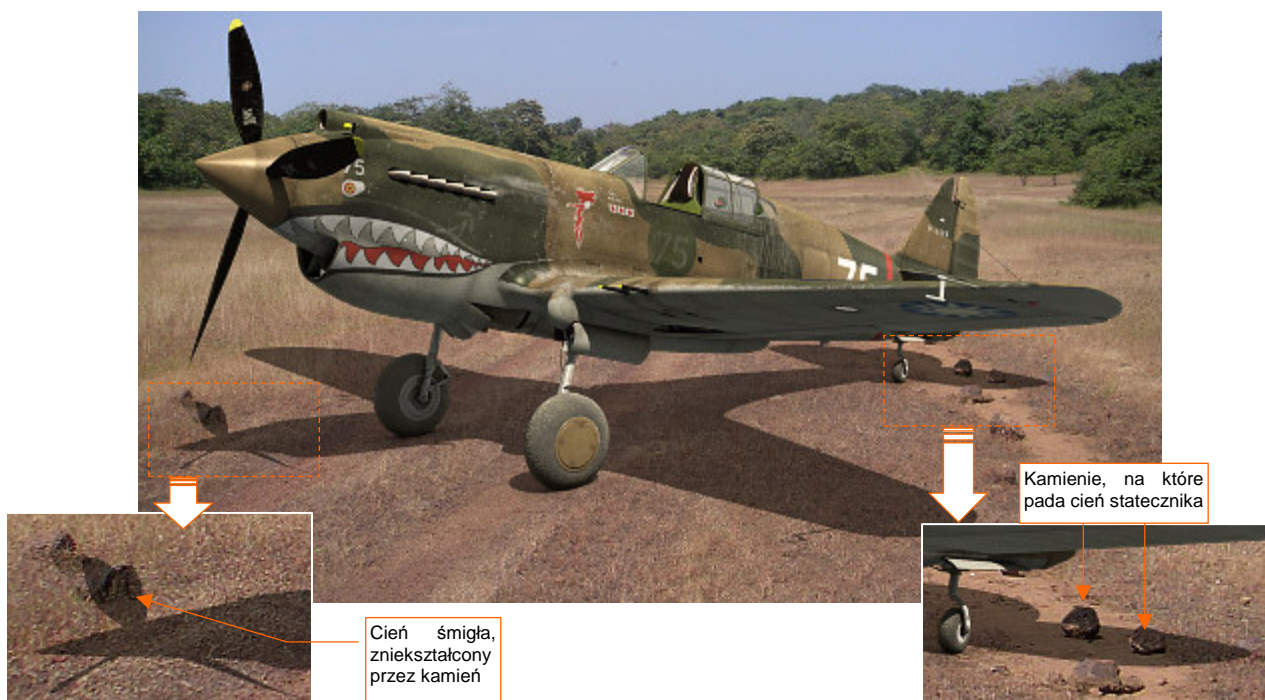
Aby lepiej dopasować zarys bryły do kształtu kamienia, trzeba czasami ją obrócić lub zmienić skalę. Wszystkie te zmiany wykonuj w widoku kamery, względem lokalnego środka obiektu.

Po umieszczeniu pierwszego kamienia sklonuj go, i ustaw ten klon pod kamień następny. Gdy już wszystkie będą rozmieszczone, przypisz siatce **Stone** nowy materiał. Powinna to być drobna przeróbka materiału podłoża (**Ground.Shadow**), polegająca na wyłączeniu opcji **Links and Pipeline:ZTransp** (Rysunek 8.2.26):



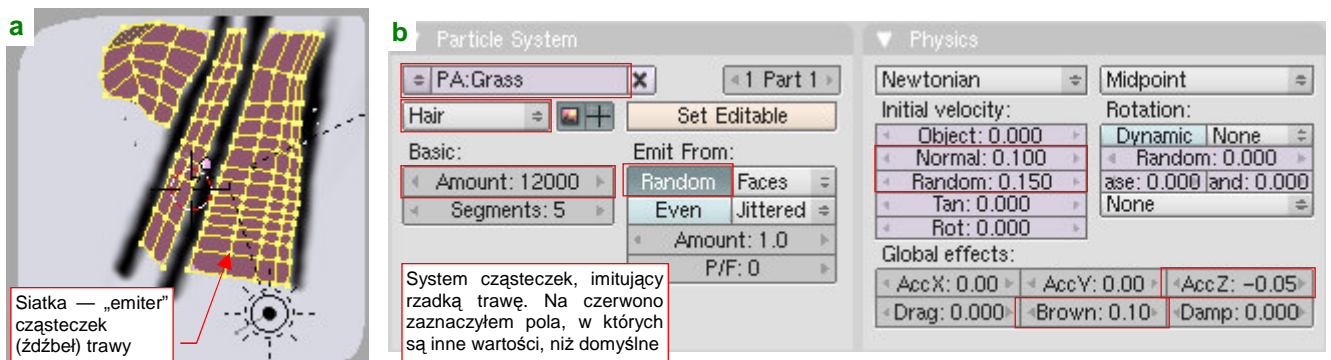
Rysunek 8.2.26 Materiał, przypisany kamieniom (**Ground.Stone**)

Nadaj temu materiałowi nazwę **Ground.Stone**. Rysunek 8.2.27 przedstawia uzyskany rezultat:



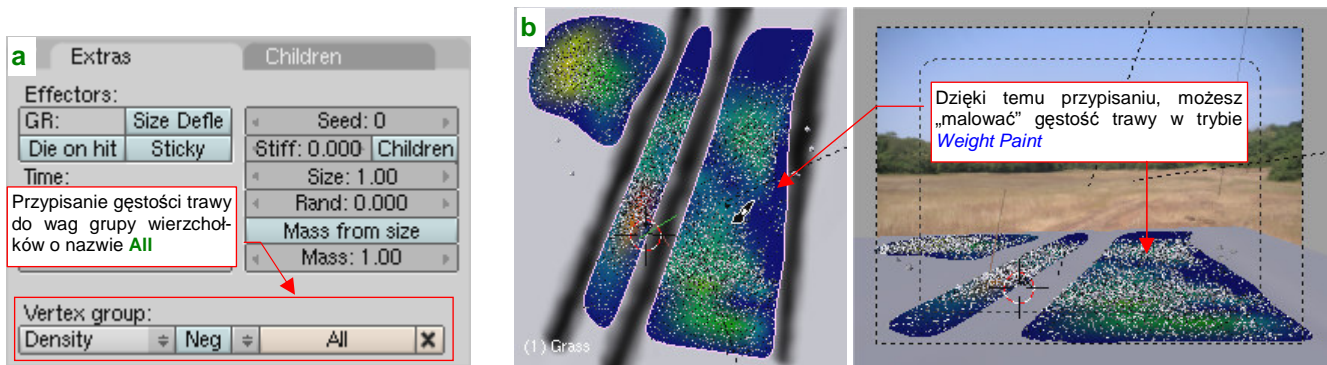
Rysunek 8.2.27 Rezultat pocieniowania kamieni

Kolejnym drobnym krokiem w kierunku realizmu będzie dodanie efektu oświetlonych i zacienionych źdźbeł trawy, widocznej na fotografii. Pełen opis, jak je przygotować, znajdziesz na str. 1040. Tutaj omówię tylko główne fazy przygotowania tego efektu. Zacznij od naniesienia na podłoże siatki (Rysunek 8.2.28a). Potem stwórz przypisany do niej system cząsteczek typu **Hair** (Rysunek 8.2.28b):



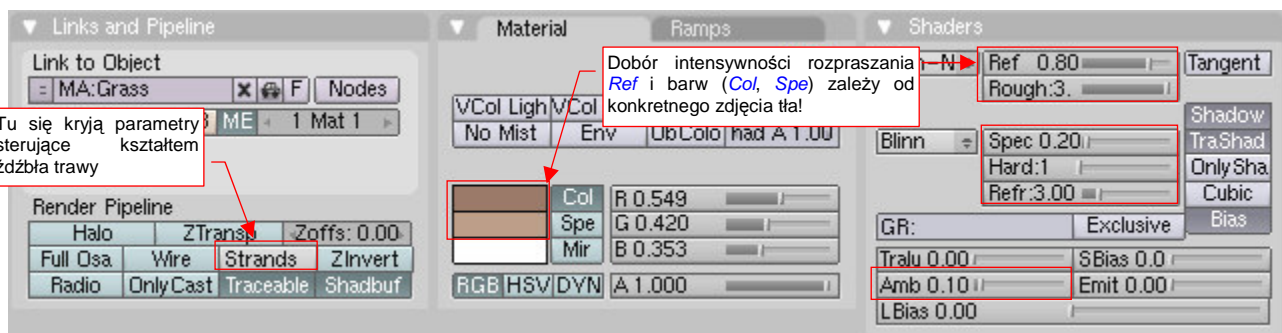
Rysunek 8.2.28 Trawa — siatka i jej system cząsteczek (**Particles**)

Wszystkie wierzchołki siatki emitującej trawę przypisałem do grupy o nazwie **All**. Powiązałem z tą grupą system cząsteczek (Rysunek 8.2.29a). Pozwala to „malować” gęstość trawy za pomocą wag (Rysunek 8.2.29b):



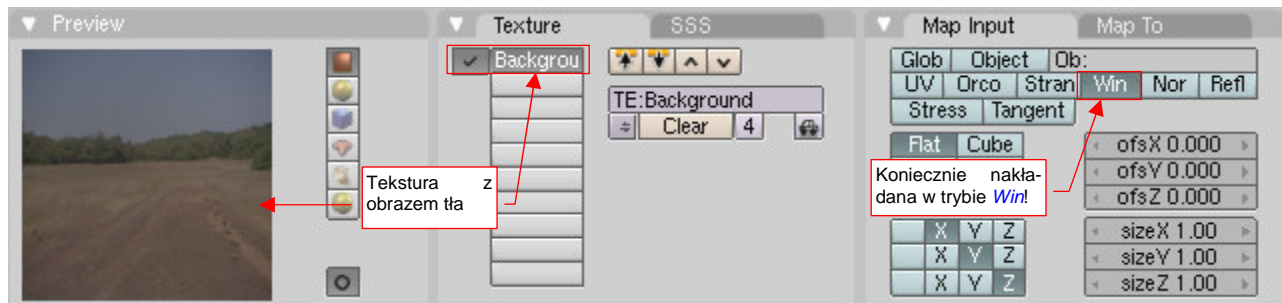
Rysunek 8.2.29 Malowanie (wag) gęstości trawy trybie (tryb **Weight Paint**)

Przypisałem trawie (a w gruncie rzeczy siatce, która ją „emituje”) materiał **Grass** (Rysunek 8.2.30):



Rysunek 8.2.30 Materiał, jakim jest pokryta trawa (**Grass**)

To materiał o minimalnej połyskliwości (**Shaders:Spec**) i niewielkim wpływie efektu AO (**Shaders:Amb**). Jego globalna barwa **Col** jest modulowana przez teksturę z obrazem tła (Rysunek 8.2.31):



Rysunek 8.2.31 Tekstura barwy materiału **Grass**

Zwróć uwagę, że ta tekstura jest nakładana prostopadłe do płaszczyzny widoku (**Map Input:Win** — por. str. 527, 528). Rysunek 8.2.32b) przedstawia render z tak przygotowaną trawą:



Rysunek 8.2.32 Porównanie: cień bez trawy i cień z trawą

Był to już ostatni ze szczegółów, którymi starałem się „urealnić” cień samolotu. Żaden z nich nie zmieniał rezultatu w sposób znaczący. A jednak, wszystkie razem sprawiły, że nasz model wydaje się teraz w pełni wiarygodnie stać na tym zdjęciu (Rysunek 8.2.33):

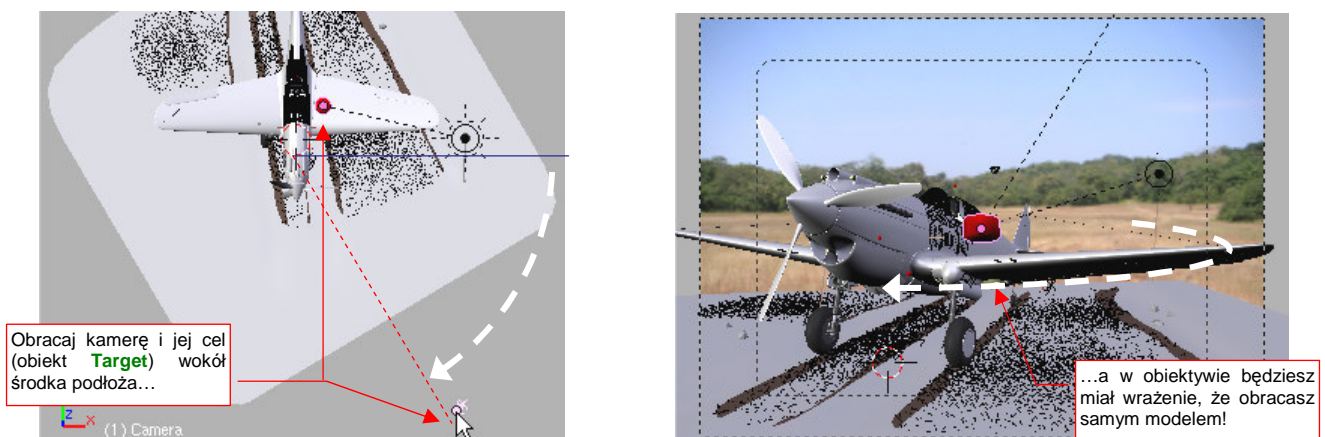


Rysunek 8.2.33 Ostateczny efekt — ukończony cień samolotu

Na pewno niejedyn z grafików komputerowych wszystkie takie poprawki cienia woli po prostu domalować na ostatecznym renderze. Można to zrobić w dowolnym programie do grafiki rastrowej (Photoshop, GIMP...). Ja jednak uważam, że lepiej przygotować podłoże w przestrzeni 3D, w oryginalnej scenie Blendera. Łatwiej później jest poprawić pozycję kamery, czy też wykonać inne zmiany w układzie sceny.

Zobacz chociażby, jak łatwo jest teraz zmienić kompozycję obrazu. Załóżmy, że chcesz ustawić samolot pod innym kątem, a tło i podłoże mają pozostać nieruchome. Pamiętaj, jak przypisaliśmy „gruntowi” ograniczenie *Locked Track*, śledzące położenie kamery (por. str. 527)? To właśnie dla szybkiego wykonania takich zmian!

Przypisz jeszcze tylko do „gruntu” wszystkie światła (relacją *Parent*). W ten sposób zapewnimy sobie ich niezmienną pozycję<sup>1</sup>. Teraz ustaw płaszczyznę widoku równoległą do podłoża (*View→Align View→Align View To Selected*). Umieść w jej środku kursor 3D (*Object→Snap→Cursor -> Selection*). Ustaw oś obrotu (*Pivot*) na *3D Cursor*. Następnie zaznacz kamerę i jej cel (obiekt *Target*). Zaczynij je obracać (Rysunek 8.2.34a):



Rysunek 8.2.34 Obracanie kamery bez zmiany pozycji podłoża

<sup>1</sup> Tylko nie zapomnij przed przypisaniem usunąć na chwilę z tych lamp ich ograniczeń *TrackTo*. (Obiekty z ograniczeniami „złe znoszą zmianę rodzica” — ulegają wtedy dziwnym przesunięciom i obrotom). Po przypisaniu do podłoża z powrotem przypisz światłom te ograniczenia, kierujące je na obiekt *Target*.

Jednocześnie obserwuj w drugim oknie, co się dzieje w obiektywie kamery. Choć obracasz tylko kamerę, to tam wygląda to wszystko tak, jak gdybyś obracał sam model! Reszta „sceny” — podłoże, oświetlenie i tło — jest w kamerze nieruchoma. I właśnie o taki efekt chodziło! W ten sposób możesz zmienić ujęcie modelu, nie tracąc czasu na dopasowania trawy, kamieni, podłoża do nieruchomej fotografii tła. Zobacz sam: jednym ruchem ustawieś nasz samolot równoległe do widocznej na zdjęciu drogi (Rysunek 8.2.35):



A może wstawić tu drugi samolot?  
I kręcących się wokół niego mechaników?

**Rysunek 8.2.35** Ujęcie z innej pozycji

Grafik, który poprawia render w Photoshopie, musiałby w takim przypadku usiąść, i ponownie poprawiać „surowy” cień samolotu. A my mamy to gotowe! Teraz można byłoby po lewej wstawić drugi samolot, a wokół niego — figurki pilota i ze dwóch mechaników... Spróbuj tego sam, bo ja muszę już ograniczać rozmiar tej książki!

Skończmy więc poprzednie ujęcie. Na ostatecznym obrazie zdecydowałem się jeszcze tylko dodać drugą maszynę (ale bez mechaników), by podkreślić, że jest to coś w rodzaju polowego lotniska (Rysunek 8.2.36):



**Rysunek 8.2.36** Ostateczna wersja sceny

Zrobiłem to metodą „dwóch plików”, opisaną w poprzedniej sekcji (por. str. 521).

Jak się chyba już przekonałeś, przygotowanie sceny na ziemi, a szczególnie na lotnisku gruntowym, jest bardziej pracochłonne niż sceny w powietrzu. Wszystko to wiąże się z „walką z cieniem” (samolotu), który trzeba jak najlepiej wkomponować w zdjęcie tła. Wyraźnie to widać chociażby we wnioskach, podsumowujących tę sekcję.

### Podsumownie

- Fotografię, którą wykorzystujemy jako tło, można przed użyciem poddać „wstępnej obróbce” w GIMP. W tym przypadku zdjęcie zostało wyostrene (*Sharpen*), by lepiej pasowało do „twardego” wykończenia detali na renderze modelu (str. 524);
- Oświetlenie sceny na ziemi składa się z 3 „lamp”. Podstawową jest słońce (*Sun* — str. 524), o odpowiednio dużej energii. Dwa pozostałe to słabe światła półsferyczne (*Hemi*). Jedno z nich rozjaśnia spód modelu, i ma wyłączone odbłyски (*No Specular*). Drugie służy do „zaakcentowania” wybranych elementów (str. 525);
- Cień samolotu uzyskujemy na powierzchni, podstawionej pod koła modelu (str. 526). To nasz model „gruntu”. Najprostszą metodą integracji cienia z tłem jest pokrycie powierzchni gruntu odpowiednią teksturą, mapowaną w trybie *Win*. Niestety, wadą tego rozwiązania jest niewielkie rozmycie obrazu tła na teksturze i brak pełnej możliwości sterowania całym zakresem jasności cienia (str. 528);
- Lepszym, choć bardziej skomplikowanym, sposobem uzyskania cienia jest wykorzystanie dwóch powierzchni, pokrytych materiałem z włączoną opcją *Only Shad(ow)* (str. 529). Pierwsza z nich ma wyłączony wpływ efektu *AO (Ambient Occlusion)* i tworzy podstawowy cień. Druga ma włączony wpływ *AO* i odpowiada za zaciemnienia wokół kół samolotu (str. 531). Intensywnością każdego z tych dwóch efektów można niezależnie sterować, zmieniając przejrzystość tych powierzchni;
- Cień samolotu można dodatkowo „urealnić”, modelując na siatce podłoża niektóre widoczne na obrazie tła nierówności. (Chodzi o te, które zniekształcają cień). Mogą to być koleiny polnej drogi (str. 530), a także kamienie (str. 532);
- Podczas pracy może Ci się przydać narzędzie do nanoszenia „linii koncepcyjnych” — *Grease Pencil* (str. 530);
- W scenach na lotniskach polowych zawsze warto odwzorować trawę, na którą pada cień samolotu. Można ją uzyskać za pomocą tzw. systemu cząsteczek (*Particles*), typu *Hair* (str. 534);
- Światła sceny warto przypisać (relacją *Parent*) do płaszczyzny podłoża. Podłoże można skierować (ograniczeniem *Locked Track*) na kamerę (str. 527). Takie ustawienia umożliwiają wygodne obracanie kamery wokół modelu (str. 535, 536);