

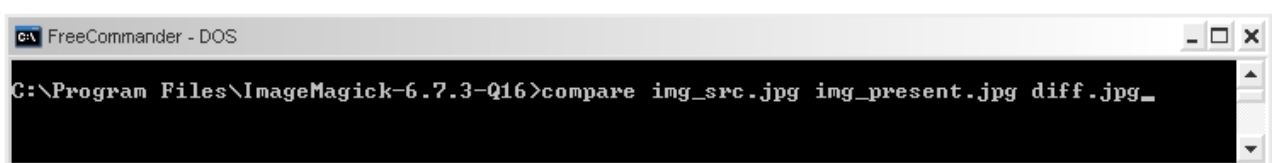
## Testowanie aplikacji internetowych – proste porównywanie obrazów

Testując aplikacje internetowe często wykorzystuje się zrzuty ekranów (Print Screen) do porównania stanu w jakim znajduje się aplikacja ze stanem oczekiwanym (najczęściej ujętym w specyfikacji). Szczególnie wtedy stosujemy takie rozwiązanie kiedy sprawdzamy nie tyle zawartość co sposób wyświetlania strony np. po wprowadzeniu nowych arkuszy stylów. Dodatkowo kiedy wykorzystujemy automatyzację do testów regresji i testujemy wyświetlanie się danej strony po poprawkach wówczas takie porównywanie obrazów przez specjalne narzędzie bardzo upraszcza nam pracę i skraca czas na poszukiwanie różnic. Zapewne niektórzy zapytają jakim narzędziem możemy robić zrzuty ekranu w testach automatycznych? Tutaj z rozwiązaniem przychodzi nam Selenium, szczególnie jeśli jest to skrypt wykorzystujący API SeleniumRC lub Selenium Webdriver (Selenium 2). Dwa proste polecenia „captureScreenshot()” lub „captureEntirePageScreenshot()” robią zrzut pulpitu lub zrzut całej strony załadowanej do przeglądarki (drugie polecenie działa jedynie w Firefoxie) i zapisują wynik do pliku.

Prostą i darmową aplikacją, którą można wykorzystać do porównywania obrazów jest ImageMagick[1]. ImageMagick jest narzędziem do tworzenia, edycji, komponowania (tworzenia), lub konwersji obrazów bitmapowych. Potrafi odczytywać i zapisywać obrazy w różnych formatach (ponad 100), w tym DPX , EXR , GIF, JPEG, JPEG-2000, PDF, PhotoCD, PNG, Postscript, SVG oraz TIFF. ImageMagick pozwala na zmianę rozmiaru, obrócenie obrazu, stworzenie lustrzanego odbicia, zniekształcenia, przycięcia i transformacje obrazu, porównanie obrazów i wiele wiele innych. Funkcjonalność ImageMagick jest zazwyczaj wykorzystywana z linii poleceń lub można używać jego funkcji z programów napisanych w swoim ulubionym języku programowania.

Po ściągnięciu ImageMagick i zainstalowaniu (choć nie jest to wymagane, przynajmniej w przypadku środowiska Windows) możemy przejść do wykonywania dostępnych funkcji z linii poleceń. W tym artykule zajmiemy się funkcją „compare”, której zadaniem jest analiza i porównywanie obrazów.

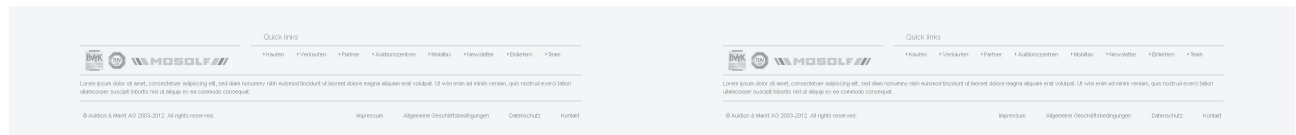
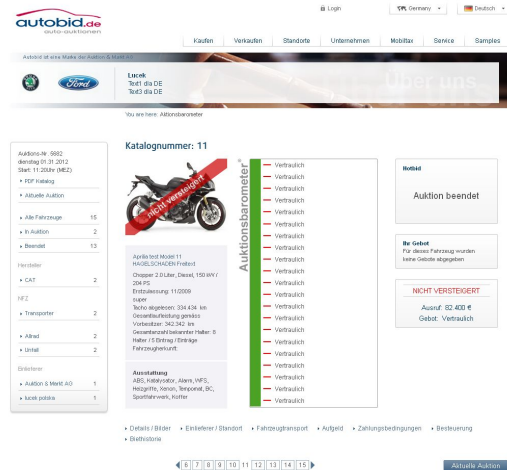
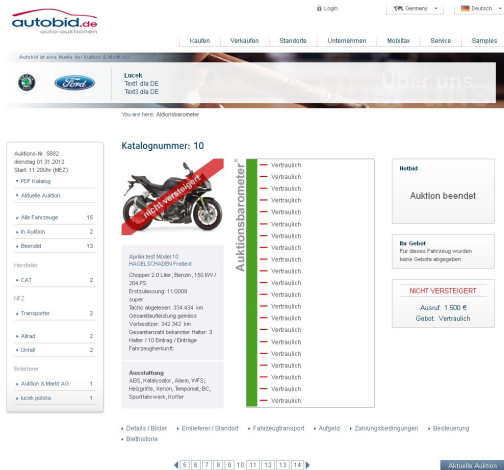
Podstawowa składnia polecenia compare przyjmuje trzy wartości, tak jak jest to przedstawione na poniższym obrazku



*Ilustracja 1: Podstawowa składnia polecenia compare*

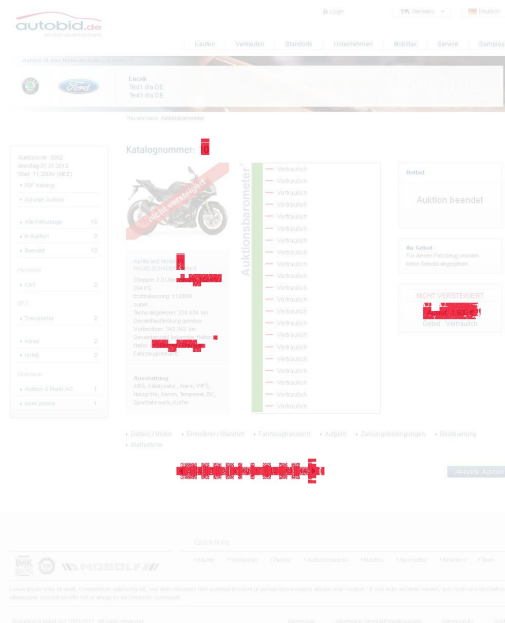
Wartość „img\_src.jpg” to obraz źródłowy, oryginał z którym porównujemy obraz obecny - „present.jpg”. Wynik, czyli różnicę tych dwóch porównywanych obrazów zapisujemy do pliku „diff.jpg”.

Poniżej znajduje się przykład porównania dwóch obrazów przedstawiających widok barometru z naszej aplikacji dla dwóch różnych pojazdów. Różnice między nimi zaznaczone są kolorem czerwonym na obrazie wynikowym.



Ilustracja 2: Pojazd numer 1

Ilustracja 3: Pojazd numer 2



Ilustracja 4: Różnice między dwoma obrazami

Jak widać na powyższym przykładzie różnice zaznaczone są kolorem czerwonym co pozwala w prosty i szybki sposób znaleźć elementy różniące się. Tych różnic w tym przypadku nie ma zbyt wiele, ograniczają się jedynie do niewielkich zmian w treści strony.

Spróbujmy zatem z innym przykładem, gdzie różnice w obrazie dotyczą już samej grafiki. Grafikę do przykładu zaczerpnąłem ze strony źródłowej projektu gdyż jest prosta i czytelna i świetnie odzwierciedla właściwości funkcji compare.



Ilustracja 6:  
img1.gif



Ilustracja 5:  
img2.gif

Powyższe obrazki to dwa slajdy z animacji. Porównajmy je teraz za pomocą funkcji `compare` wydając polecenie

```
comapre img1.gif img2.gif diff.gif
```

w rezultacie otrzymamy poniższy obraz, który dosadnie pokazuje różnice



Ilustracja 7: Porównanie obrazów `img1.gif` i `img2.gif`

Funkcja `compare` posiada wiele dodatkowych parametrów, które w mniejszym bądź większym stopniu mogą przydać się w procesie porównywania obrazów, dodatkowo parametry te można ze sobą łączyć w jednym zapytaniu. Poniżej przedstawię kilka przykładów kombinacji parametrów, które możemy zastosować w tym poleceniu.

Zmieńmy zatem kolor pokazywanych różnic z czerwonego na czarny

```
compare img1.gif img2.gif -highlight-color Black diff.gif
```



Jeśli nie chcemy mieć „cienia” drugiego obrazka wówczas zastosujemy parametr „`-compose Src`”

```
compare img1.gif img2.gif -compose Src diff.gif
```



Aby wygenerować maskę w skali szarości zmienionych pikseli zastosujemy polecenie z trzema parametrami

```
compare img1.gif img2.gif -compose src -highlight-color white -lowlight-color  
black diff.gif
```



Jak widać możliwości porównywania obrazów za pomocą ImageMagick są bardzo duże. W artykule ograniczyłem się jedynie do prostych poleceń ale dzięki dużej liczbie parametrów tej funkcji każdy będzie mógł dopasować swój wynikowy obraz do swoich potrzeb. Osoby zainteresowane tymi parametrami odsyłam na stronę <http://www.imagemagick.org/script/compare.php#options>, gdzie są one wyszczególnione. Ogromną zaletą tego programu jest niewątpliwie fakt, że jest on udostępniany za darmo jako wolne oprogramowanie zarówno jako binarna dystrybucja gotowa do uruchomienia lub jako źródło, które można w dowolny sposób kopiować i modyfikować.

[1] - <http://www.imagemagick.org>