

Konwersja do odcieni szarości

Autorzy: Kacper Kapuściak, Hubert Mazur, Konrad Perłowski

Opis projektu

Celem projektu jest napisanie programu pozwalającego zamieniać zdjęcie kolorowe na zdjęcie czarno-białe metodą mieszania kanałów, innymi słowy użytkownik ma możliwość wyboru ilości informacji z kanału czerwonego, zielonego i niebieskiego, które mają pozostać w końcowym zdjęciu. Ponadto program umożliwia pozostawienie jednej wybranej barwy i jej odcieni na zdjęciu, które w takiej sytuacji już nie będzie czarno białe. Dodatkowo istnieje możliwość pozostawienia wybranych intensywnych barw na zdjęciu czarno-białym.

Projekt robiliśmy wspólnie, wszystkie z wyżej wymienionych funkcji opracowywaliśmy razem spotykając się we trójkę. Poza wymaganiami projektu Hubert zrobił możliwość cofnięcia oraz wykonania ponownie zrobionej zmiany, Kacper zajął się testami, a Konrad dokumentacją.

Założenia wstępne przyjęte w realizacji projektu

Program powinien pozwolić na wybór proporcji pomiędzy mieszanymi kanałami wykorzystując trzy suwaki z dozwolonymi wartościami od -200% do 200%. Dla zdjęcia czarno-białego program powinien umożliwić zastosowanie bichromii. Wszystkie naniesione zmiany powinny być widoczne na podglądzie generowanym w czasie rzeczywistym (dłuższy bok generowanego podglądu nie przekracza 500px). Zmodyfikowane zdjęcie można zapisać na dysk.

Analiza projektu

Zdefiniowanie struktur danych

Oryginalne zdjęcie oraz jego kopie przetrzymywane są w obiekcie klasy `wxImage`. Wszystkie zmiany naniesione na zdjęcie przechowywane są w `std::vector`.

Specyfikacja interfejsu użytkownika

Użytkownik ma możliwość wyboru pozostawienia koloru na zdjęciu czarno-białym za pomocą trzech górnych suwaków, ma do wyboru ustawienie każdego z suwaków na wartości od 0 do 255.

Przy pomocy kolejnych trzech suwaków użytkownik ma możliwość edytowania czarno-białego zdjęcia poprzez wybór wartości poszczególnych kanałów.

Poniżej suwaków znajdują się przycisk dzięki któremu użytkownik ma możliwość wyboru koloru, który zostanie użyty przy bichromii.

Użytkownik ma również opcje zapisu edytowanego zdjęcia poprzez opcję "save image" dostępną w menu `file`. Istnieje również opcja zapisu ustawionych parametrów poprzez opcję `save parameters`. Parametry zostaną zapisane do pliku `dane.xd`, a przy wyborze opcji `load parameters` zostanie one wczytane z tego pliku.

Decyzja o wyborze narzędzi programistycznych

Do naszego projektu zdecydowaliśmy się użyć biblioteki wxWidgets, ponieważ ta biblioteka wspiera bardzo wygodny w budowaniu aplikacji wxFormBuilder oraz posiada przyjazny dla programisty interfejs. Do kompilacji użyliśmy kompilatora g++ w wersji 8.3.0. W zadaniu używaliśmy standardu c++11. Sam projekt robiliśmy przy użyciu programu Clion.

Podział pracy i analiza czasowa

Większość projektu zrobiliśmy wspólnie. Na początku stworzyliśmy projekt w wxFormBuilder. Dodaliśmy wszystkie potrzebne przyciski i suwaki. Następnie zajęliśmy się implementacją konwersji do odcieni szarości metodą mieszania kanałów. Posiadając już możliwość zamiany zdjęcia kolorowego na czarno-białe postanowiliśmy zająć się bichromią, a następnie możliwością zachowania wybranego koloru w czarno-białym obrazie. Na samym końcu dodaliśmy możliwość zapisu obrazu na dysk.

Opracowanie niezbędnych algorytmów

Algorytm użyty przy konwersji do odcieni szarości użyty w metodzie `pixelGreyscale` polega na pobraniu wartości RGB z piksela o współrzędnych podanych w argumentach funkcji oraz pomnożeniu ich przez wartość ustawioną na suwakach. Następnie z otrzymanych wartości liczymy średnią arytmetyczną i wynik ustawiamy na każdą ze składowych RGB.

Algorytm użyty do bichromii polega na pomnożeniu składowych RGB wybranego koloru przez wartość obliczoną w algorytmie konwersji do odcieni szarości i podzieleniu otrzymanej liczby przez 255.

Algorytm służący do skalowania obrazka polega na sprawdzeniu czy którykolwiek z boków ma więcej niż 500 pikseli i jeśli tak to ustawienie dłuższego boku na 500 pikseli a drugiego na wartość taką aby została zachowana oryginalna proporcja.

Kodowanie

Opis klas

- `MyApp` jest to klasa niezbędna do uruchomienia programu
- `GUI` jest to klasa wygenerowana przez program wxFormBuilder, znajdują się w niej wszystkie elementy interfejsu, przyciski, suwaki itp.
- `GUIframe` jest to klasa odpowiedzialna za wszystkie metody pozwalające na działanie programu
- `SliderData` jest to klasa przechowująca informacje o wprowadzonych zmianach

Opis zmiennych

- `wxImage *_image` wskaźnik przechowujący oryginał wczytanego zdjęcia
- `wxImage *_resized_image` wskaźnik przechowujący pomniejszone zdjęcie
- `wxImage *_greyed_image` wskaźnik przechowujący pomniejszone, zmienione na odcień szarości zdjęcie

- `std::vector<SliderData> _undoHistory` wektor przechowujący historię zmian nałożonych na zdjęcie
- `int _undoHistory_current_pos` zmienna przechowująca aktualną pozycję w wektorze zmian, przy wykonaniu Undo jej wartość zmniejsza się, przy Redo- zwiększa się o jeden.

Opis funkcji

- `onOpen` metoda odpowiedzialna za poprawne wczytanie zdjęcia. Wczytywane zdjęcie zapisywane jest to wskaźnika `_image`, następnie tworzona jest kopia obrazka, która jest odpowiednio zmniejszana i przychowywana we wskaźniku `_resized_image`, dodatkowo zostaje utworzona druga kopia obrazka `_greyed_image`, która jest używana w innych metodach
- `onSave` metoda służy do zapisywania zedytowanego zdjęcia. Funkcja ta wywołuje metodę `colorKeepingGreyscale` przekazując jej oryginał w celu naniesienia na niego wszystkich ustawionych zmian
- `onLoadParams` metoda służąca do wczytania zapisanych parametrów z zapisanego pliku `.xd`.
- `onSaveParams` metoda służąca do zapisu ustawionych przez użytkownika parametrów jako plik `data.xd`.
- `refresh` metoda rysująca podgląd zdjęcia na środku ekranu
- `channelGreyscale` metoda konwertująca `_resized_image` na zdjęcie czarno białe poprzez wywołanie metody `pixelGreyscale` na każdym pixelu zdjęcia
- `pixelGreyscale` metoda przyjmująca współrzędne pixela, którego chcemy kolor chcemy zmienić. Zmiana koloru polega na użyciu metody mieszania kanałów korzystając z wartości ustawionych na suwakach. Metoda ta również odpowiedzialna jest za nałożenie efektu bichromii kiedy jest ona włączona przez użytkownika
- `channelKeepingGreyscale` metoda ta służy do konwersji zdjęcia na czarno-białe z zachowaniem wybranych wyrazistych kolorów. Metoda ta używa wartości ustawionych na suwakach i jeżeli dany kolor jest obecny na zdjęciu to jest on zachowany, w innym przypadku jest on zamieniany na odcień szarości przy użyciu metody `pixelGreyscale`
- `onSlider...` metody wywołujące metody służące do konwertowania zdjęcia na czarno-białe (`channelGreyscale`, `pixelGreyscale`, `channelKeepingGreyscale`)
- `pushUndoHistory` wywołanie metody sprawia zapisanie parametrów konwersji do wektora dokonanych zmian
- `setSliders` metoda ustawia pozycję suwaków
- `onUndo` Metoda wywoływana przy cofaniu naniesionych zmian, ustawia wszystkie parametry zgodnie z zapisanymi na poprzedniej pozycji w wektorze zmian (poprzedniej względem aktualnej).
- `onRedo` Metoda wywoływana przy wykonywaniu zmian ponownie, po uprzednim cofnięciu. Wykorzystuje parametry zapisane w wektorze zmian na pozycji następnej względem aktualnej.

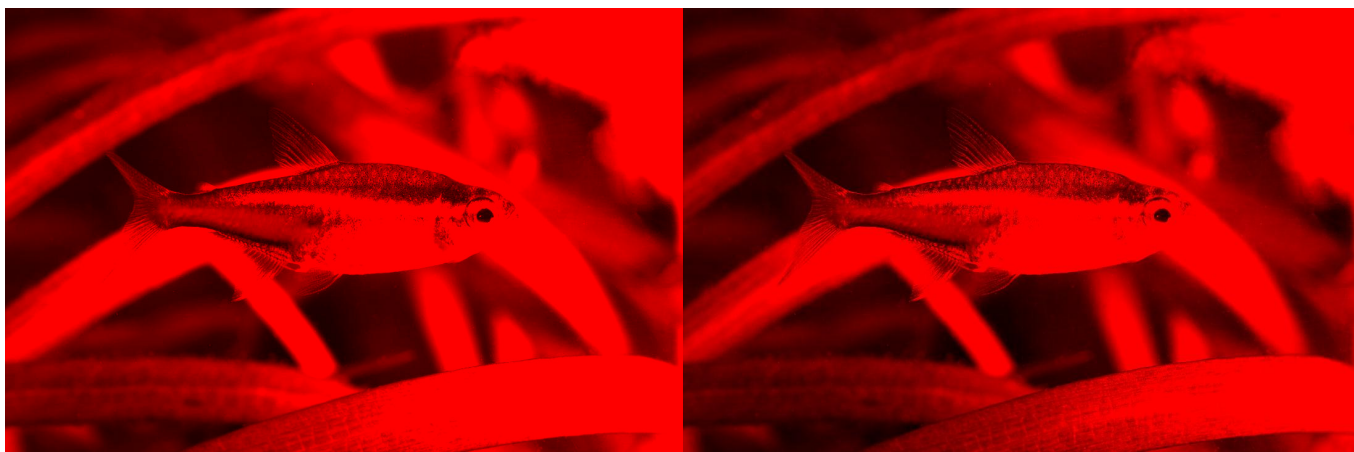
Testowanie

Działanie naszego programu testowaliśmy manualnie porównując wyniki operacji na zdjęciu z wynikami otrzymanymi w programie GIMP.

Po ustawieniu wartości suwaków odpowiedzialnych za mieszanie kanałów na wartości 200% oraz ustawienie suwaków w programie GIMP na wartości 2 otrzymujemy (po lewej stronie obraz otrzymany w programie graficznym GIMP, po prawej w naszym):



Podobnie sprawdziliśmy działanie bichromii, porównanie naszego wyniku z wynikiem w GIMPie (po lewej stronie obraz otrzymany w programie graficznym GIMP, po prawej w naszym):



Pozostałe opcje sprawdziliśmy bez porównywania z innymi programami, testowaliśmy działanie funkcji zapisania i wczytania obrazu, które działają bezbłędnie.

Wdrożenie, raport i wnioski

W programie udało się wykonać wszystkie wymagania podstawowe, możliwe jest konwertowanie zdjęcia na czarno-białe zachowując wybraną przez użytkownika ilość informacji z poszczególnych kanałów. Możliwym również jest zastosowanie bichromii i zapis edytowanego zdjęcia. Elementem do poprawy na pewno jest optymalizacja, program nie działa zadowalająco płynnie.

Z wymagań rozszerzonych udało nam się zaimplementować opcję pozostawienia wybranego wyrazistego koloru na zdjęciu czarno-białym oraz zapis ustawionych parametrów w programie. Pozostawienie wybranego koloru na zdjęciu nie działa idealnie i na pewno można tę funkcję poprawić w przyszłości.