

Projektowanie Oprogramowania Systemów

Wymagania projektu

Klient:

Użytkownikiem końcowym tworzonego oprogramowania, jest osoba szukająca programu do przetwarzania obrazów. Jako użytkownicy, brane są pod uwagę osoby prywatne oraz firmy wykorzystujące tego typu oprogramowanie.

Wymaganie funkcjonalne:

- wczytywanie obrazów ze ścieżki umieszczonej w pliku ".ini"
- zapisywanie przetworzonych obrazów do ścieżki z pliku ".ini"
- wyrównywanie jasności wczytanych obrazów
- tworzenie mozaiki miniatur obrazków przed oraz po przetworzeniu

Wymagania poza-funkcjonalne:

- wyświetlanie mozaik w dwóch oknach w celu szybkiego sprawdzenia wyniku przetwarzania
- zastosowanie przetwarzania wielowątkowego w celu przyspieszenia pracy programu

Scenariusz użycia:

1. Użytkownik wprowadza do pliku „.ini” ścieżki wskazujące na pliki, które chce wprowadzić do programu.
2. Użytkownik wprowadza do pliku „.ini” ścieżkę do której mają być zapisane obrazy wynikowe
3. Użytkownik uruchamia program.
4. Program wczytuje obrazy oraz wyrównuje ich jasność.
5. Program wyświetla w dwóch oknach mozaiki miniaturek obrazów.
6. Użytkownik może obejrzeć wyświetlone mozaiki oraz obrazy wynikowe umieszczone w ścieżce zapisu.

Dodatkowe biblioteki i programy:

Do wykonania tego zadania należało użyć dodatkowych bibliotek takich jak: opencv oraz INIParser. Opencv służy do obróbki obrazu a INIParser do pobierania danych z plików typu „ini”.

Do wykonania tego projektu nie było konieczne użycie dodatkowych narzędzi. Wobec tego użyto tylko takich narzędzi jak: Visual Studio 2015, Microsoft Visio, GIT, GitHub, Doxygen. (Bugzilla nie była potrzebna, ponieważ jest już zawarta w GitHub). Zamiast programu Doxygen można było użyć programu Sandcastle. Jednak ostatecznie zdecydowano się na Doxygen.

Specyfikacja techniczna:

Wykonany programu służy do obróbki cyfrowej obrazu. Program jest uruchamiany z linii poleceń. W pliku images.ini znajdują się ścieżki obrazów, które mają być przetworzone oraz adres folderu, do którego program później zapisze nowe obrazy powstałe po przetworzeniu.

Program dokonuje wyrównania jasności odczytanych obrazów. W folderze docelowym jest zapisana matryca miniatur plików źródłowych oraz matryca miniatur obrazów docelowych (jeden plik powinien zawierać sklejone ze sobą zmniejszone obrazy źródłowe a drugi plik – sklejone obrazy po przetworzeniu, aby można szybko było zobaczyć efekty działania programu).

Użyto takich narzędzi jak: Visual Studio 2015, Microsoft Visio, GIT, GitHub, Doxygen; oraz dodatkowych bibliotek: opencv i INIParser.

Harmonogram pracy:

Kwiecień 2016

- 29.04.2016 Rozpoczęcie projektu. Rozdzielenie zdań między osobami w grupie.

Maj 2016

- 1.05.2016 Założenie repozytorium plików na GitHubie
- Instalowanie odpowiednich programów (Git, Visual Studio) oraz bibliotek (kompilacja biblioteki OpenCv)
- Wybranie i sprawdzenie metody wyrównania jasności obrazów

Czerwiec 2016

- Prace nad kodem źródłowym. Napisanie funkcji do:
 - analizy plików .ini i wczytywania obrazów
 - przetwarzania obrazów
 - tworzenia mozaik z obrazów i wyświetlania ich na ekranie
 - zapisywania przetworzonych obrazów
- Połączenie poszczególnych funkcjonalności w jeden program
- Wykonanie dokumentacji w programie Doxygen
- Wykonanie modelu UML w programie Visio.