

# *Porządkowanie zdjęć*

Przemysław Rodzik

Anna Szumilas

Wiktoria Zaczek



# AGH

Każda osoba w równej części zajmowała się kodem oraz opracowaniem dokumentacji

# 1. Opis Projektu

W dobie fotografii cyfrowej, duże ilości zdjęć często przechowywane są na płytach CD lub DVD. Reorganizacja takiego archiwum, zwłaszcza przy znacznej liczbie płyt jest dość kłopotliwa. Celem projektu jest napisanie programu, który na podstawie zawartości płyt CD lub DVD odtworzy na dysku twardym istniejącą na nich strukturę katalogów, a następnie wypełni je miniaturkami oryginalnych zdjęć. Dzięki takiemu zgromadzeniu miniatur wszystkich zdjęć w jednym miejscu łatwiej będzie zorientować się w posiadanych zbiorach.

## 2. Założenia wstępne przyjęte w realizacji projektu

*Wymagania podstawowe:*

- Program pozwala na wybieranie płyty CD/DVD lub dowolnego katalogu na dysku oraz wygenerowanie do katalogu docelowego miniatur nie przekraczających zadanych wymiarów.
- Użytkownik może ustalać maksymalną wysokość oraz szerokość miniaturki oddzielnie i jednocześnie, przy czym miniaturki powinny zachować proporcje boków oryginału.
- Miniaturki zapisywane są w formacie JPG z ustalonym przez użytkownika stopniem kompresji.
- Program wczytuje zdjęcia w formacie JPG i BMP
- Program liczy zdjęcia przed rozpoczęciem tworzenia miniatur i w trakcie pracy informuje użytkownika o procentowej ilości wykonanych miniatur.
- Puste katalogi lub zawierające tylko nieobsługiwane formaty są ignorowane w trakcie tworzenia struktury katalogów.

*Wymagania rozszerzone:*

- Program obsługuje formaty natywne aparatów cyfrowych (RAW) oraz PNG i TIFF
- Program działa w trybie półautomatycznym; miniaturki są prezentowane na ekranie, w tym czasie użytkownik może lewym klawiszem kursora obrócić zdjęcie w lewo o 90 stopni, prawym w prawo o 90 stopni, a po naciśnięciu górnego klawisza kursora następuje zapis miniaturki i przejście do następnego zdjęcia.
- Program umieszcza dodatkowy plik graficzny, które zawierają "miniaturki miniaturowe", znajduje się tam 40 miniatur (5 x 8) prezentujących zdjęcia z danego katalogu.

## 3. Analiza projektu

- *Specyfikacja danych wejściowych:*  
Dane wejściowe to foldery z obrazkami.  
Formaty obrazków wejściowych: png, jpg, raw, bmp, tiff

- *Opis oczekiwanych danych wyjściowych:*  
Folder z miniaturkami w formacie jpg z podanym stopniem kompresji
- *Zdefiniowanie struktur danych*  
Głównie korzystaliśmy z obiektów biblioteki wxWidgets.
- *Specyfikacja interfejsu użytkownika*  
Przycisk “Wczytaj plik” - otwiera okno dialogowe z wyborem folderu z którego będą wczytywane zdjęcia  
Przycisk “Eksportuj” - kompresuje załadowane do aplikacji zdjęcia i zapisuje je w formacie jpg we wskazanym folderze  
Pola “Szerokość” i “Wysokość” - pozwalają na zaznaczenie po czym odbywana jest kompresja i wpisanie wartości w pikselach do jakiej obrazek ma być zamieniony  
Slider “Kompresja” - pozwala wybrać wielkość kompresji, od wartości min do max  
Checkbox “Tryb półautomatyczny” - pozwala na wybranie trybu półautomatycznego  
Panel wyświetlania - wyświetlają się na nim obrazki w trybie półautomatycznym
- *Wyodrębnienie i zdefiniowanie zadań*  
Funkcje w programie dzielą się na parę głównych kategorii:
  - wygląd aplikacji
  - przeszukiwanie folderów
  - zapisywanie obrazu do aplikacji
  - konwertowanie obrazu
- *Decyzja o wyborze narzędzi programistycznych*  
Korzystaliśmy z Visual Studio 2019 do pisania i kompilowania programu oraz z wxWidgets i wxFormBuildera do stworzenia szkicu aplikacji oraz z biblioteki Freeimage.

## 4. Podział pracy i analiza czasowa

- ❖ Interfejs użytkownika przy użyciu wxFormBuilder - Przemysław Rodzik (6h)
- ❖ Generowanie miniatur o zadanych rozmiarach - Anna Szumilas (3h)
- ❖ Wczytywanie formatów plików (bez Raw) - Wiktoria Zaczek (3h)
- ❖ Zliczanie zdjęć - Przemysław Rodzik (1h)
- ❖ Sprawdzanie katalogów - Wiktoria Zaczek (2h)
- ❖ Zajęcie się strukturą katalogów - Wiktoria Zaczek, Przemysław Rodzik (3h)
- ❖ Eksportowanie plików - Anna Szumilas (2h)
- ❖ Tryb półautomatyczny - Anna Szumilas (4h)
- ❖ Ścieżki do obrazów - Przemysław Rodzik (1h)
- ❖ Dokumentacja - Anna Szumilas, Wiktoria Zaczek, Przemysław Rodzik (3h)

## 5.Opracowanie i opis niezbędnych algorytmów

Funkcje wczytujące obrazy przeszukują wszystkie podkatalogi folderu i dodają je do tablicy. Wobec tego korzystają z algorytmu poprzecznego przeszukiwania drzewa binarnego, sprawdzane czy w poddrzewie znajdują się katalogi. Korzystamy z podstawowej funkcji do kompresji zdjęć z biblioteki FreeImage.

## 6.Kodowanie

virtual void \_LoadFolder(wxCommandEvent& event) - wczytuje folder ze zdjęciami

virtual void \_Export(wxCommandEvent& event) - eksportuje skompresowany obrazek

virtual void \_UpdateInterface(wxUpdateUIEvent& event) - odświeża ekran interfejsu

void \_KeyPressed(wxKeyEvent& event) - służy do rozpoznawania który przycisk został wciśnięty podczas trybu półautomatycznego

virtual void \_WidthCheckBox(wxCommandEvent& event) - uwzględnia kompresję według szerokości

virtual void \_HeightCheckBox(wxCommandEvent& event) - uwzględnia kompresję według wysokości

virtual void \_SemiAutomaticMode(wxCommandEvent& event) - wyświetla obrazy w trybie półautomatycznym podczas konwersji

*Funkcje do manualnego ustawiania wartości kompresji:*

- virtual void \_MaxHeightSpinCtrl(wxSpinEvent& event)
- virtual void \_MaxHeightSpinCtrlText(wxCommandEvent& event)
- virtual void \_MaxHeightTextEnter(wxCommandEvent& event)
- virtual void \_MaxWidthSpinCtrl(wxSpinEvent& event)
- virtual void \_MaxWidthSpinCtrlText(wxCommandEvent& event)
- virtual void \_MaxWidthTextEnter(wxCommandEvent& event)
- virtual void \_CompressionLevel(wxCommandEvent& event)

void \_Repaint() - odświeża ekran interfejsu

void \_GetPaths(const wxDir& dir) - zapisuje ścieżki dostępu do plików w tablicy, wykorzystuje rekurencyjne szukanie podfolderów

const wxString \_GetAllFiles(const wxDir& dir, const wxString extension) const - zwraca listę wszystkich nazw zdjęć

void \_NextImage() - przełączanie do następnego obrazu w trybie półautomatycznym

void \_SaveOne(int index); - wykorzystywane w trybie półautomatycznym, zapisuje skompresowane zdjęcie do folderu docelowego.

void \_AddToMiniatures(FIBITMAP\* miniatures, FIBITMAP\* bitmap, int& widthIndex, int& heightIndex); - dodaje kolejną miniaturkę do pliku

`void _CopyDir(wxString& source, wxString& target);` - tworzy strukturę folderu źródłowego w folderze docelowym

`bool _ImageCheck(wxString& currPath) const;` - sprawdza czy dany plik jest wspierany przez naszą aplikację

## 7. Testowanie

Program został przetestowany przez skład niezależnych osób. Składowe funkcje zostały przetestowane ze stałymi znanymi wartościami poza programem, bez interfejsu graficznego. Po uprzednim testach napotkaliśmy problem jednoczesnego wczytywania i zapisywania zdjęć z tego samego folderu. To zostało uwzględnione odpowiednimi wyjątkami. Kolejnym napotkanym problemem było wczytywanie plików RAW. Po rozeznaniu się w sytuacji okazało się że do obsługi plików raz jest potrzebna specjalna biblioteka i wspólnie podjęliśmy decyzję o rozwoju aplikacji w tym kierunku w późniejszym czasie.

## 8. Wdrożenie, raport i wnioski

Prawidłowo zostały wykonane wszystkie podstawowe oraz dodatkowe wymagania projektu. Exe działa prawidłowo, ale nie da się wczytać formatu RAW. W przyszłości można naprawić obsługę plików RAW.