

Sprawozdanie 1

Temat:

„Obsługa skanera płaskiego (TWAIN lub WIA)”

Skład grupy:

Kamil Gondek 263916

Małgorzata Andrasz 264473

Prowadzący: dr inż. Jarosław Mierzwa

Spis treści

1) Cel laboratorium.....	3
2) Wstęp Teoretyczny.....	4
a) Skaner Płaski	4
• CCD	4
• CIS	5
• LiDE	6
b) Sterownik WIA.....	7
c) Format JPEG.....	7
d) Format PNG	7
e) Format TIFF.....	7
f) Format BMP.....	7
3) Opis zadania do wykonania	8
4) Przedstawienie aplikacji.....	8
a) Użyte biblioteki.....	8
b) Interfejs graficzny	8
c) Omówienie najważniejszych fragmentów kodu	13
5) Przedstawienie działania aplikacji	15
6) Wnioski	19
7) Referencje.....	20

1) Cel laboratorium

Celem laboratorium było zapoznanie się z zasadą działania skaner płaskiego z wykorzystaniem sterownika WIA. Konfiguracją sterownika by możliwe było skanowanie monochromatyczne jak i kolorowe oraz modyfikacja rozmiarów skanu. Sposobem zapisu plików graficznych do formatu JPG, PNG, TIFF, BMP. Należało opracować aplikację w języku C# która będzie umożliwiała zeskanowanie dokumentu. Wyświetlenie skanu na ekranie z możliwością rotacji pliku graficznego oraz zapisaniem do wybranego formatu.

2) Wstęp Teoretyczny

a) Skaner Płaski

Skaner to urządzenie peryferyjne, umożliwiające przetworzenie obrazu dwuwymiarowego lub trójwymiarowego do postaci cyfrowej. Mówiąc prościej, zadaniem skanera jest przeniesienie danych z kartki papieru do komputera, poprzez stworzenie elektronicznej wersji określonego dokumentu. Wygenerowany format cyfrowy można swobodnie przeglądać i edytować na komputerze.

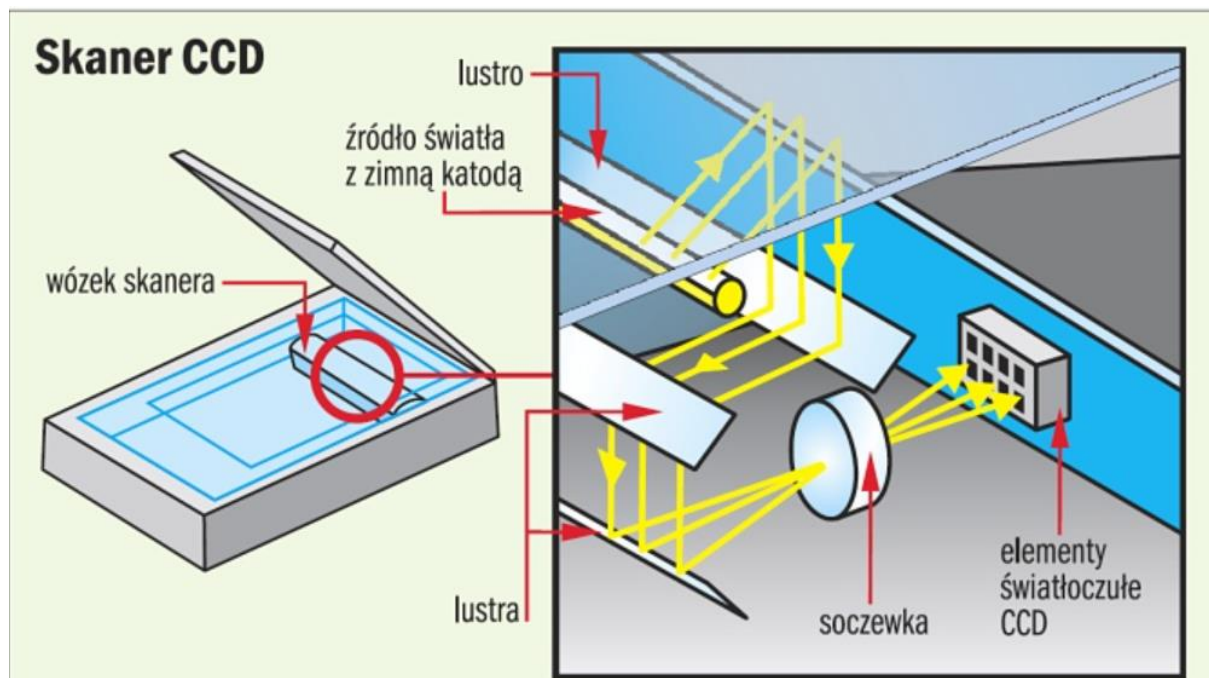
Możemy je podzielić ze względu na budowę na skanery:

- CCD

W skanerach CCD używa się do oświetlenia skanowanego dokumentu podłużnej, przypominającej świetlówkę lampy z zimną katodą (jej długość odpowiada szerokości obszaru skanowania). Emituje ona intensywne światło, zbliżone barwą do naturalnego. Lampa znajdująca się tuż pod szybą skanera, na której kładzie się materiał do skanowania, jest przymocowana do specjalnego wózka napędzanego silnikiem krokowym. Dzięki temu źródło światła może znaleźć się bezpośrednio pod każdym fragmentem skanowanego dokumentu, zapewniając jego równomierne oświetlenie.

Odbite od kartki światło trafia do układu optycznego (zespołu pryzmatów i soczewek), zamocowanego na tym samym wózku, na którym znajduje się lampa. Obraz przetwarzany jest linia po linii – jeden krok silnika przesuwającego lampę wraz z układem optycznym pod kartką odpowiada jednej linii zeskanowanego obrazu.

Po przejściu przez soczewki światło niosące informację o obrazie trafia na przetwornik CCD. Tam obraz jest zamieniany na postać cyfrową, a dzięki filtrom RGB (czerwony, zielony, niebieski) – ułożonym naprzemiennie nad czujnikami przetwornika – zyskuje kolory.



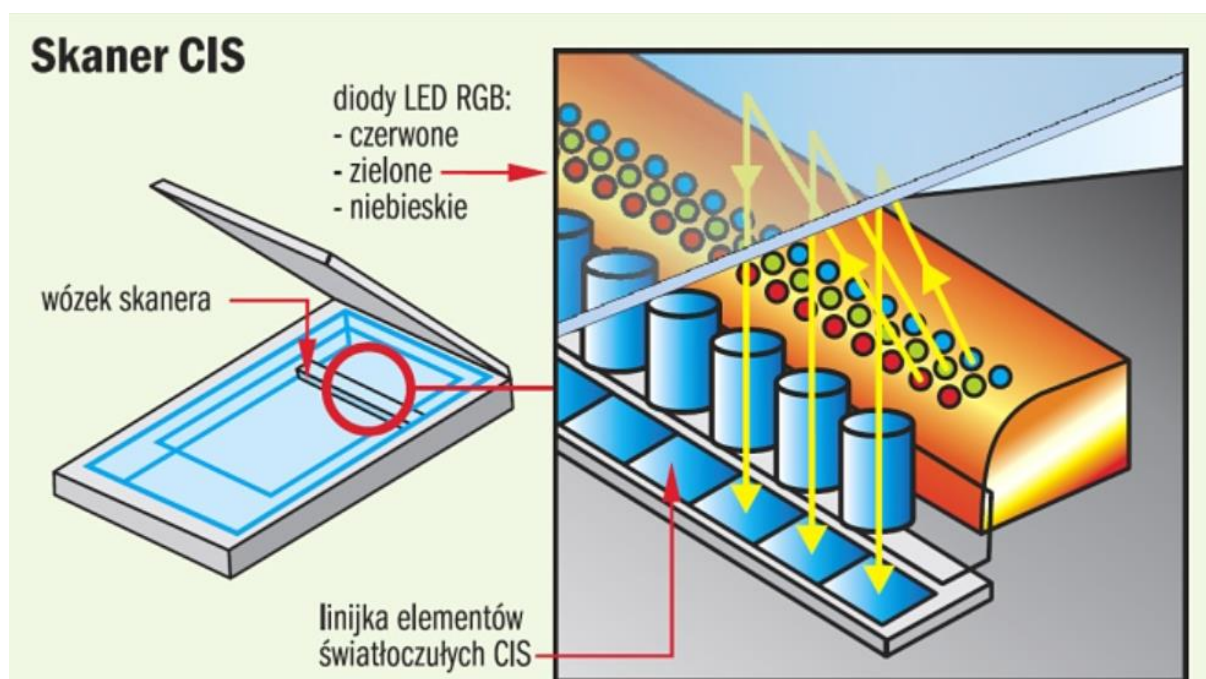
OBRAZ 2 – BUDOWA SKANERA CCD

- CIS

Technologia CIS bazuje na miniaturowych diodach LED emitujących światło w trzech podstawowych kolorach (zielonym, niebieskim oraz czerwonym) i oświetlających skanowaną kartkę. Diody umieszczone są na wspólnej listwie, przesuwającej się pod dokumentem tak jak lampa w skanerach CCD. Tuż obok diod LED na listwie znajdują się elementy światłoczułe, na które trafiają promienie światła odbitego od skanowanego oryginału.

W skanerach CIS skanowanie przebiega w skali 1:1. Nie ma też układu optycznego, bo przetworniki obrazu są tuż obok źródła światła. Technologia CIS jest tańsza w produkcji niż CCD – stosunkowo drogą optykę i lampę zastąpiły tanie elementy półprzewodnikowe. Do zasilania listwy półprzewodnikowej nie potrzeba dużo energii, więc skaner CIS może być zasilany przez złącze USB i nie wymaga, tak jak CCD, oddzielnego zasilacza. Korzystając z modeli CIS, nie trzeba też czekać na rozgrzanie się lampy – skaner taki jest od razu gotowy do pracy.

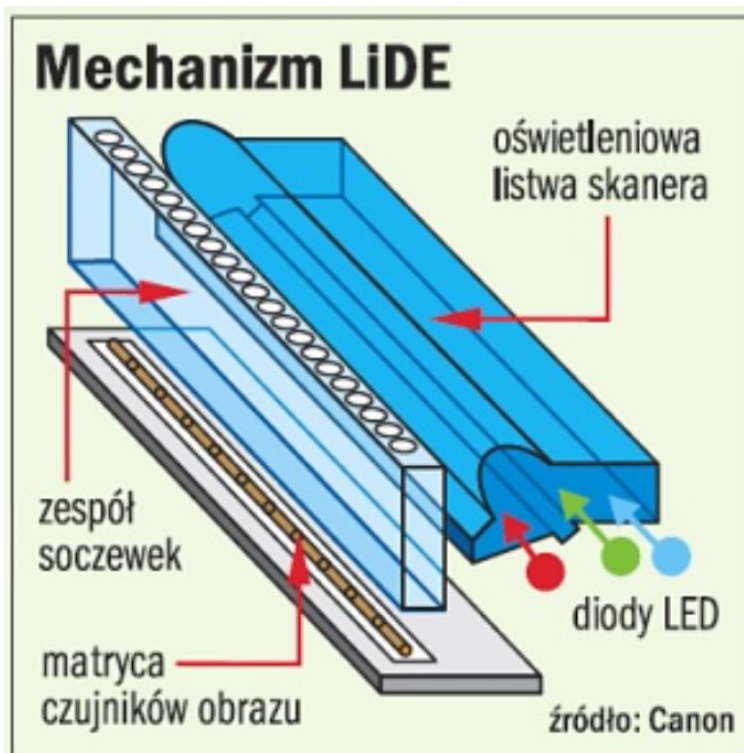
Brak lampy sprawia, że skanery CIS mają małe rozmiary i są bardzo lekkie – mogą być niewiele grubsze i niewiele cięższe od typowego zeszytu A4, podczas gdy CCD mają grubość sporej książki i ważą zwykle ponad półtora kilograma, a czasem nawet 5–6 kg. Niestety, skanery CIS mają też poważne wady.



OBRAZ 1 – BUDOWA SKANERA CIS

- LiDE

Technologia LiDE (LED inDirectExposure) łączy zalety konstrukcji CIS i CCD. W skanerach Canon LiDE źródłem światła są trzy diody LED (czerwona, zielona i niebieska). System światłowodów doprowadza światło do zespołu miniaturowych soczewek i przez nie oświetla skanowany dokument. Soczewki oraz przykryte zestawem mikrosoczewek czujniki obrazu, na które trafia odbite od skanowanego obiektu światło umieszczono na wspólnej listwie tak jak w skanerach CIS. Efektem ma być lepsza jakość obrazu niż ta otrzymywana za pomocą skanerów CIS, ale trochę gorsza od znanej ze skanów CCD. Skanery LiDE zużywają niewiele prądu więc mogą być zasilane bezpośrednio ze złącza USB.



OBRAZ 3 – BUDOWA SKANERA LiDE

b) Sterownik WIA

Windows Image Acquisition (WIA; czasami nazywany również Windows Imaging Architecture) to model sterownika Microsoft i interfejs programowania aplikacji (API) dla Microsoft Windows Me i nowszych systemów operacyjnych Windows, który umożliwia oprogramowaniu graficznemu komunikację ze sprzętem do przetwarzania obrazu, takim jak skanery, aparaty cyfrowe i cyfrowy sprzęt wideo. Po raz pierwszy został wprowadzony w 2000 roku jako część systemu Windows Me i nadal jest standardowym urządzeniem do przetwarzania obrazu i modelem API w kolejnych wersjach systemu Windows. Jest zaimplementowany jako usługa na żądanie w systemie Windows XP i nowszych systemach operacyjnych Windows.

c) Format JPEG

JPEG (ang. Joint Photographic Experts Group) – algorytm[1][2] stratnej kompresji grafiki rastrowej, wykorzystany w formacie plików graficznych o tej samej nazwie. Motywacją do powstania tego standardu było ujednolicenie algorytmów kompresji obrazów monochromatycznych i kolorowych.

Format plików JPEG/JFIF obok formatów GIF i PNG jest najczęściej stosowanym formatem grafiki na stronach WWW.

d) Format PNG

PNG (ang. Portable Network Graphics) – rastrowy format plików graficznych oraz system bezstratnej kompresji danych graficznych.

PNG został opracowany jako następca GIF w 1995 roku po ogłoszeniu przez Unisys oraz CompuServe roszczeń patentowych dotyczących kompresji LZW używanej w formatach GIF oraz TIFF. Format PNG zalecany jest przez konsorcjum W3C jako preferowany format grafiki dla sieci WWW. Dotyczy to takich obszarów jak choćby grafika prezentacyjna czy schematy blokowe.

e) Format TIFF

TIFF (ang. Tagged Image File Format) – format plików graficznych.

Format opracowany w 1986 roku przez Aldus Corporation, w porozumieniu z Microsoft i Hewlett-Packard. Prawa do formatu należą obecnie do spółki Adobe Systems, w których posiadanie weszła, poprzez przejęcie korporacji Aldus. Format ten powstał z myślą o wykorzystaniu przez PostScript, pierwotnie dla programu PageMaker.

Jest to de facto standard w DTP.

Pliki zapisane w tym formacie mają rozszerzenie „.tif” lub „.tiff”.

Format TIFF przechowuje informacje o kanałach alfa, ścieżkach, profilu kolorów, komentarzach, umożliwia także zapisywanie dokumentów wielostronicowych. TIFF pozwala na kompresję LZW, CCITT, PackBits i ZIP. Od rewizji TIFF 6.0 możliwa jest kompresja JPEG.

f) Format BMP

BMP – format pliku z grafiką bitmapową. Opracowany pierwotnie dla systemu OS/2, wykorzystywany później także w interfejsach systemów z rodziny Microsoft Windows™, jednak jako wolny od patentów jest dostępny i – mimo dużych rozmiarów – popularny jako format przechowywania danych również na wszystkich pozostałych platformach. Zawiera w sobie prostą kompresję bezstratną RLE (która nie musi być użyta), informację o użytych kolorach. Obsługuje tryby RGB oraz RGBA.

3) Opis zadania do wykonania

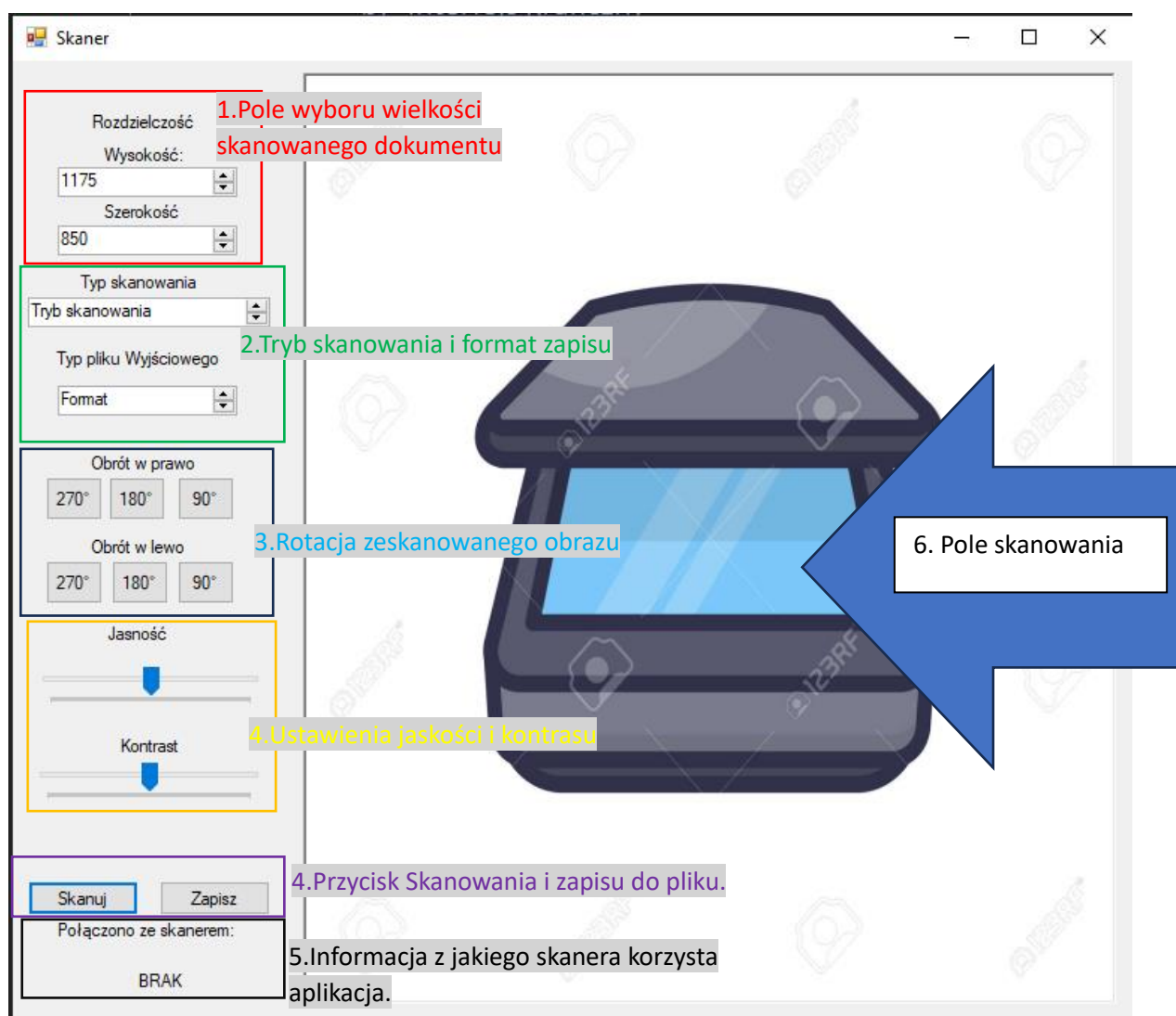
Należało wykonać aplikację w języku C# która pozwoli wybrać odpowiednie parametry skanowania takie jak tryb skanowania – kolorowe/monochromatyczne, wymiary skanowanego obrazu. Następnie po naciśnięciu odpowiedniego guzika zeskanuje dokument i wyświetli go na ekranie. Dodatkowo aplikacja powinna pozwolić na rotację skanu w prawo i lewo. Aplikacja musi również pozwolić zapisać plik w dowolnym wybranym formacie

4) Przedstawienie aplikacji

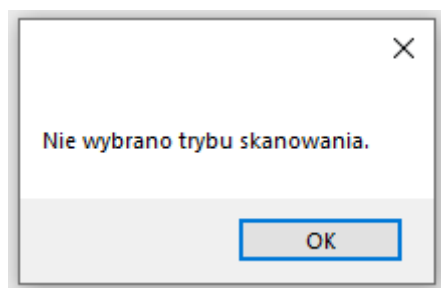
a) Użyte biblioteki

```
using System;  
using System.Drawing; - Wykorzystywana do operacji na obrazie  
using System.IO; - Wykorzystywana do operacji na plikach  
using System.Windows.Forms;  
using WIA; - Wykorzystywana do obsługi skanera
```

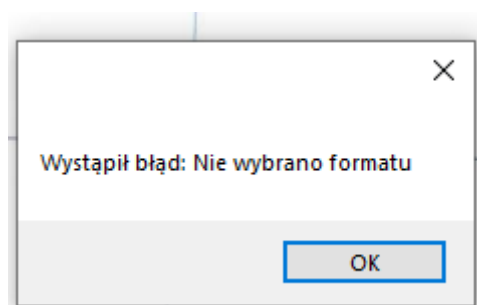
b) Interfejs graficzny



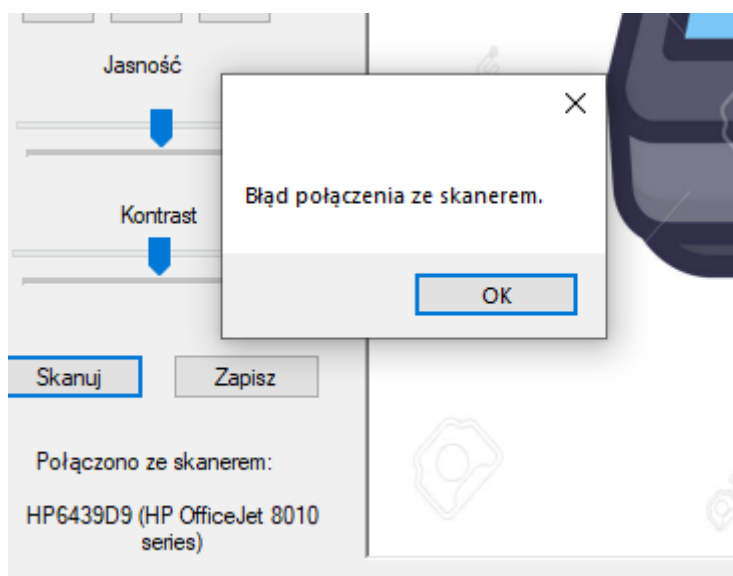
OBRAZ 4 - EKRAN GŁÓWNY APLIKACJI



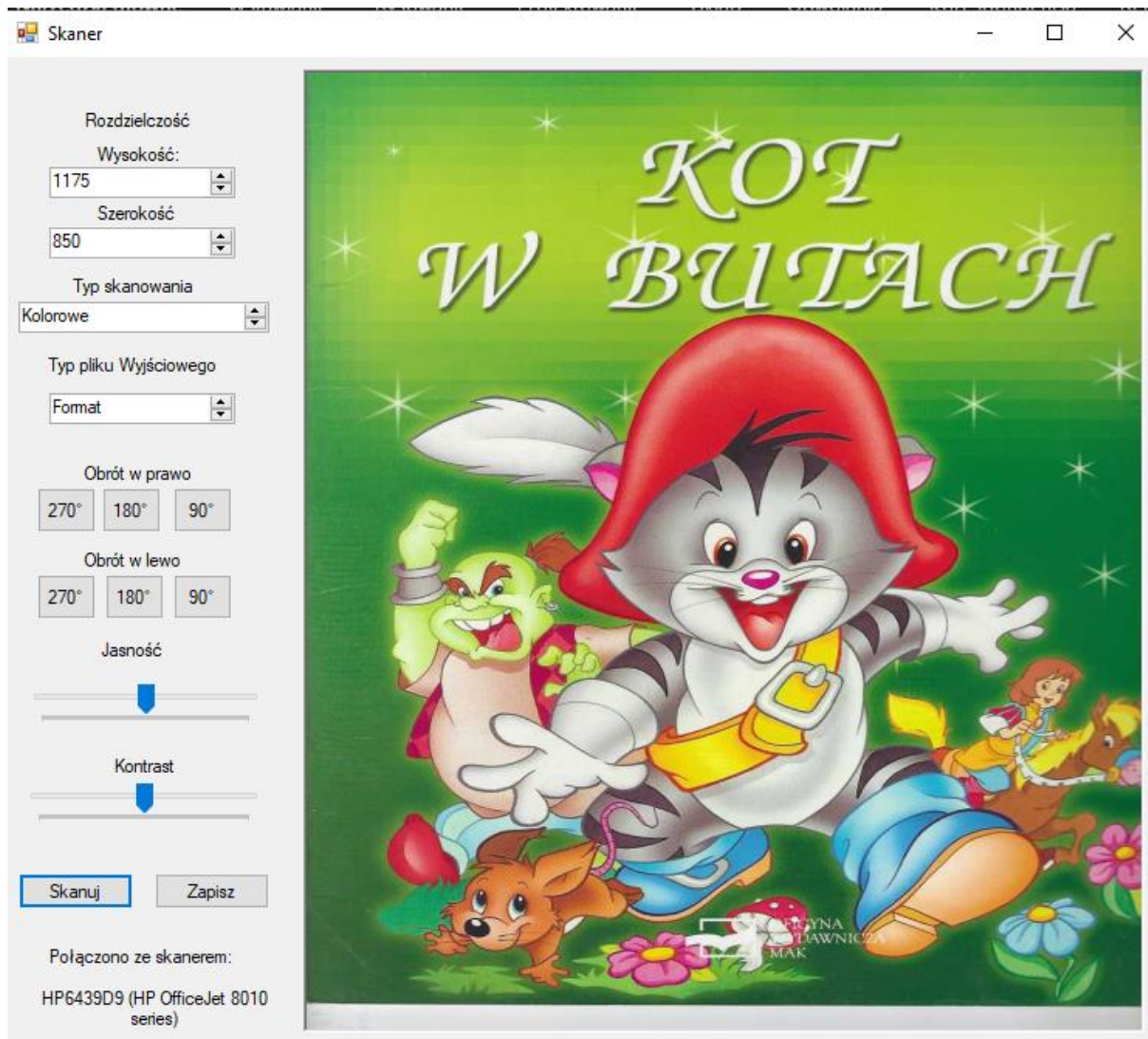
OBRAZ 5 – KOMUNIKAT O NIE WYBRANIU TRYBU SKANOWANIA.



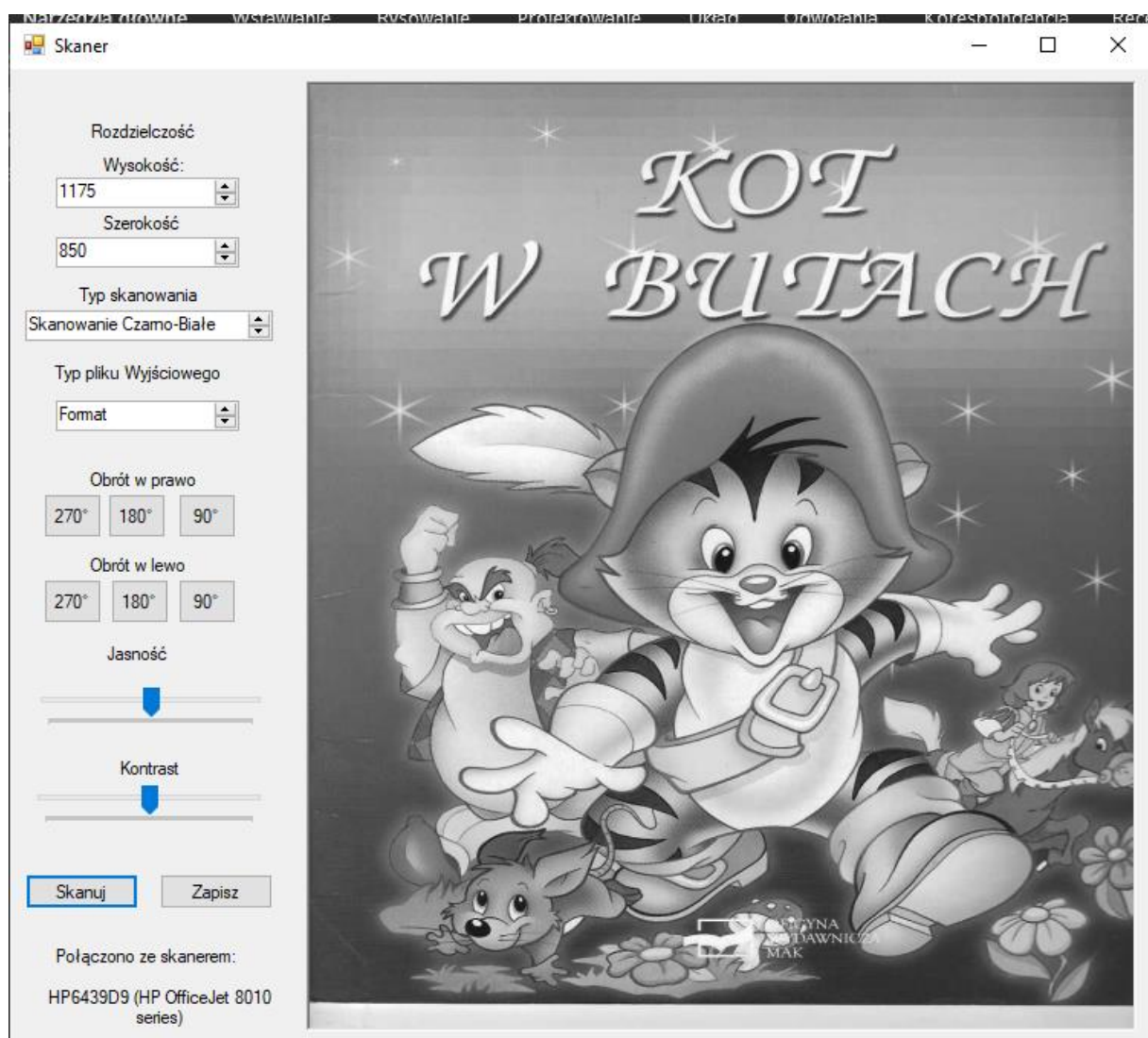
OBRAZ 6 – KOMUNIKAT O NIE WYBRANIU FORMATU ZAPISU PLIKU.



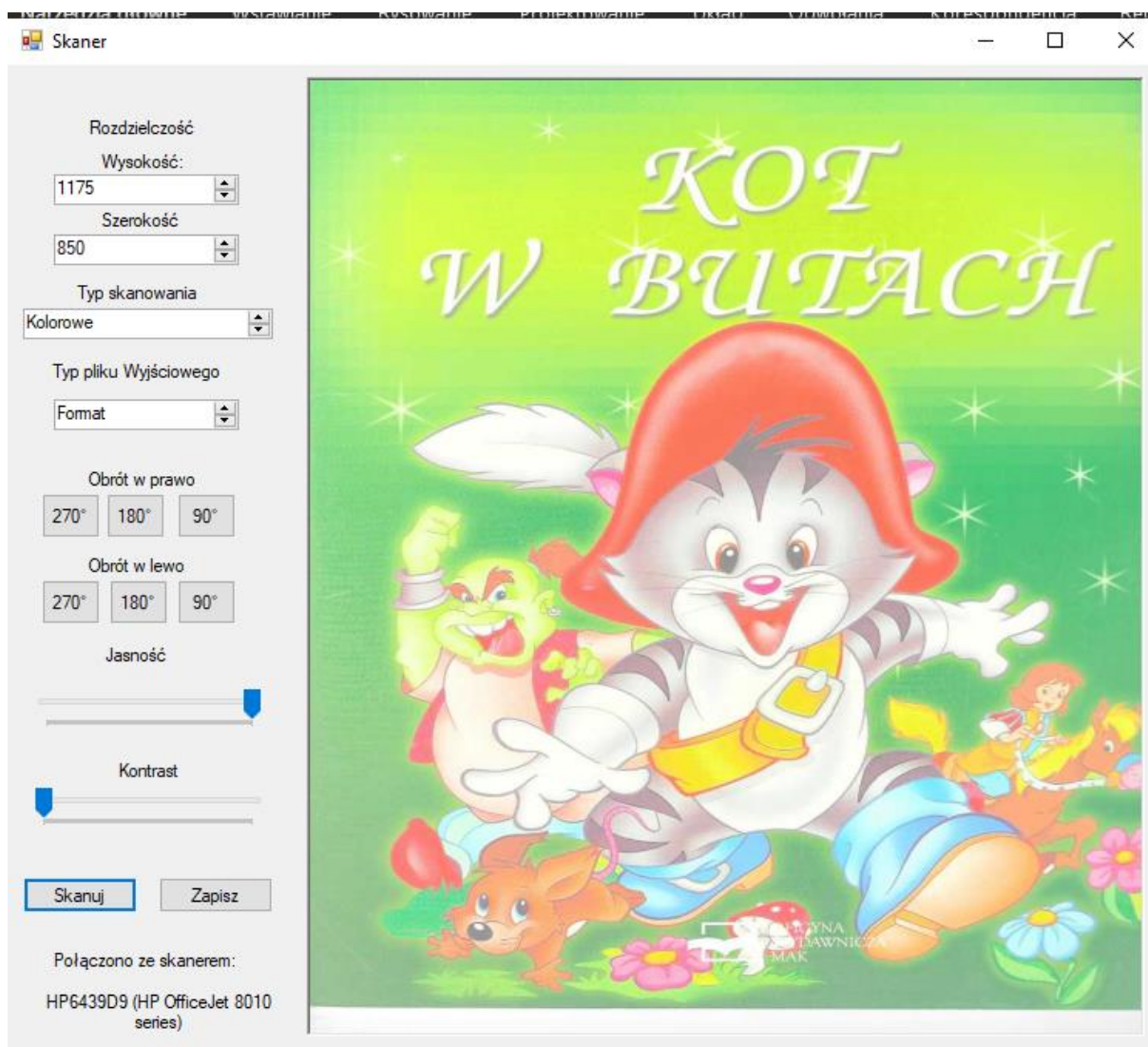
OBRAZ 7 – KOMUNIKAT O BŁĘDZIE POŁĄCZENIA ZE SKANEREM.



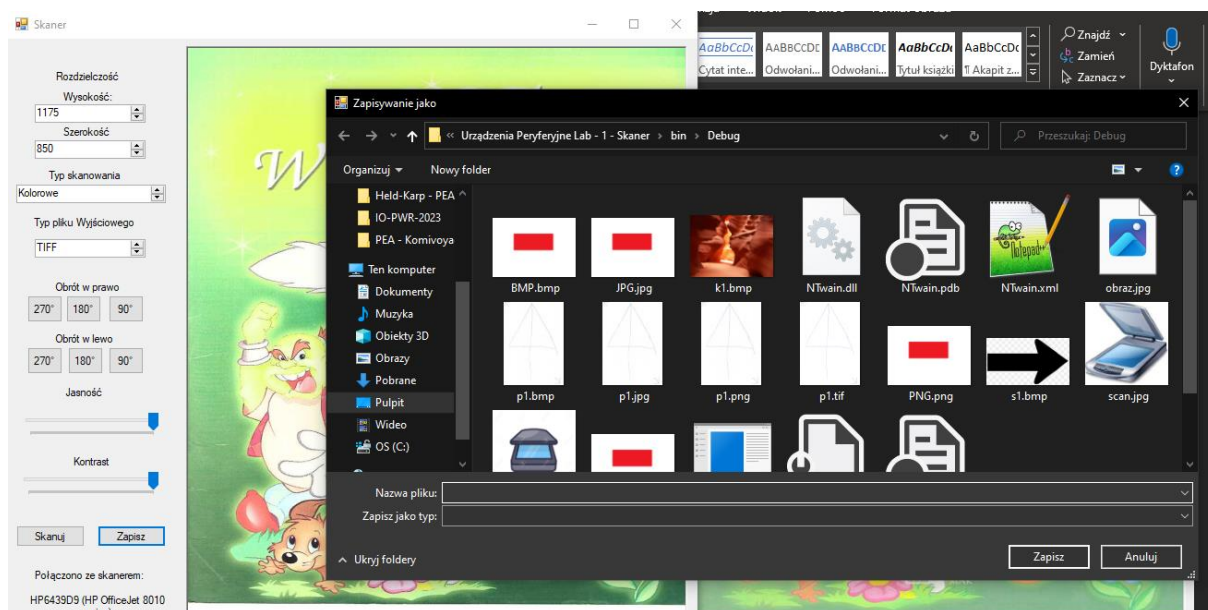
OBRAZ 8 – SKAN KOLOROWY



OBRAZ 9 – SKAN CZARNOBIAŁY



OBRAZ 10 – SKAN KOLOROWY ZE ZMODYFIKOWANĄ WARTOŚCIĄ KONTRASTU I JASNOŚCI



OBRAZ 11 – OKNO ZAPISU OBRZU DO PLIKU

c) Omówienie najważniejszych fragmentów kodu

Funkcja konwersji zmiennej ImageFile do zmiennej Image do wyświetlenia w programie.

```
1 //Konwersja
2 static Image ConvertImageFileToImage(ImageFile imageFile)
3 {
4     // Pobierz dane obrazu jako tablicę bajtów
5     byte[] imageBytes = (byte[])imageFile.FileData.get_BinaryData();
6
7     // Utwórz strumień pamięci na podstawie tablicy bajtów
8     using (MemoryStream stream = new MemoryStream(imageBytes))
9     {
10        // Utwórz obiekt Image na podstawie strumienia
11        Image obraz = Image.FromStream(stream);
12        return obraz;
13    }
14 }
```

Funkcja zapisu skanu do pliku.

```
//zapis do pliku
static void SaveImageFile(ImageFile imageFile, string filePath)
{
    // Uzyskaj dane obrazu jako tablicę bajtów
    byte[] imageBytes = (byte[])imageFile.FileData.get_BinaryData();

    // Zapisz tablicę bajtów do pliku
    System.IO.File.WriteAllBytes(filePath, imageBytes);
}
```

Funkcja rotacji obrazu.

```
private void buttonRR_90_Click(object sender, EventArgs e)
{
    if(obraz != null)
    {
        obraz.RotateFlip(RotateFlipType.Rotate90FlipNone);
        pictureBoxScan.Image = obraz;
    }
}
```

Funkcja przeszukująca wszystkie urządzenia w poszukiwaniu skanera.

```
DeviceManager deviceManager = new DeviceManager();
DeviceInfo firstScanner = null;
foreach (DeviceInfo deviceInfo in deviceManager.DeviceInfos)
{
    if (deviceInfo.Type == WiaDeviceType.ScannerDeviceType)
    {
        firstScanner = deviceInfo;
        break;
    }
}
```

Połączenie ze skanerem i utworzenie okna dialogowego.

```
1
2 Console.WriteLine("Znaleziono skaner: " + firstScanner.Properties["Name"].get_Value());
3 SkanerName.Text = firstScanner.Properties["Name"].get_Value();
4
5 device = firstScanner.Connect();
6 scannerItem = device.Items[1] as WIA.Item;
```

Konfiguracja najważniejszych parametrów skanera takich jak punkt startowy skanowania, DPI, wymiary, jasność i kontrast.

```
1
2 switch (color) {
3     case 0:
4         scannerItem.Properties["6146"].set_Value(2); // Czarno-Białe
5         break;
6     case 1:
7         scannerItem.Properties["6146"].set_Value(1); // Kolorowe
8         break;
9 }
10
11 scannerItem.Properties["Horizontal Start Position"].set_Value(1);
12 scannerItem.Properties["Horizontal Start Position"].set_Value(1);
13 scannerItem.Properties["Horizontal Resolution"].set_Value(100);
14 scannerItem.Properties["Vertical Resolution"].set_Value(100);
15 scannerItem.Properties["Horizontal Extent"].set_Value(width);
16 scannerItem.Properties["Vertical Extent"].set_Value(hight);
17 scannerItem.Properties["Format"].set_Value(wiaFormatBMP);
18 scannerItem.Properties["Brightness"].set_Value(brightness);
19 scannerItem.Properties["Contrast"].set_Value(contrast);
20
```

Skanowanie i zapis do pliku zmiennej ImageFile w formacie JPG oraz konwersja do zmiennej Image

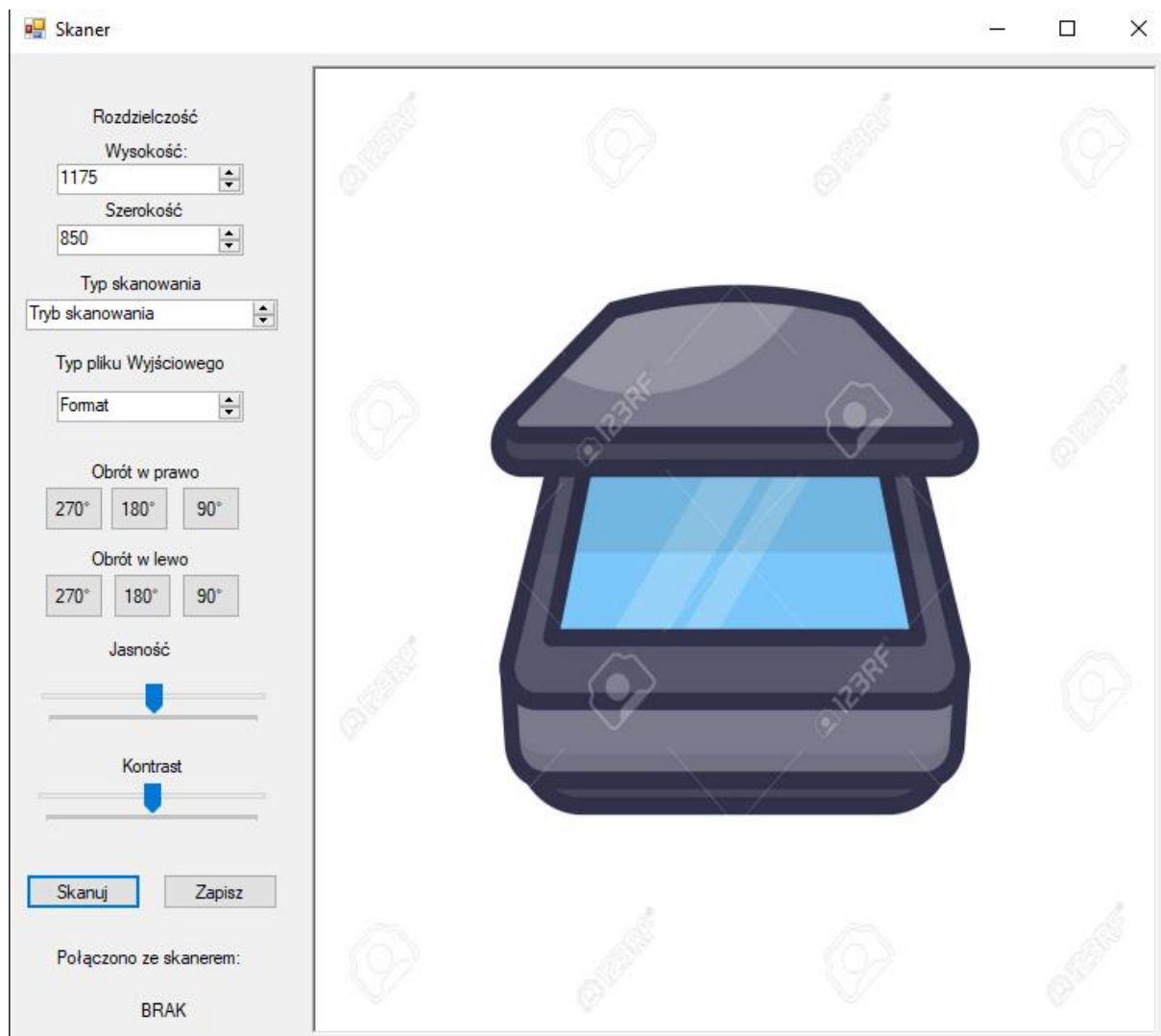
```
1
2 imgFile = (ImageFile)scannerItem.Transfer(wiaFormatJPG);
3 obraz = ConvertImageFileToImage(imgFile);
```

Zapis do pliku na przykładzie JPG – pobranie ścieżki, dodanie rozszerzenia do ścieżki, skan do właściwego formatu, zapisanie pliku.

```
1
2 case 0://JPG
3 if (saveFileDialog.ShowDialog() == DialogResult.OK)
4 {
5     string sciezkaDoPliku = saveFileDialog.FileName;
6     sciezkaDoPliku = sciezkaDoPliku + ".jpg";
7     imgFile = (ImageFile)scannerItem.Transfer(wiaFormatJPG);
8     SaveImageFile(imgFile, sciezkaDoPliku);
9
10    MessageBox.Show("Obraz zapisany");
11 }
```

5) Przedstawienie działania aplikacji

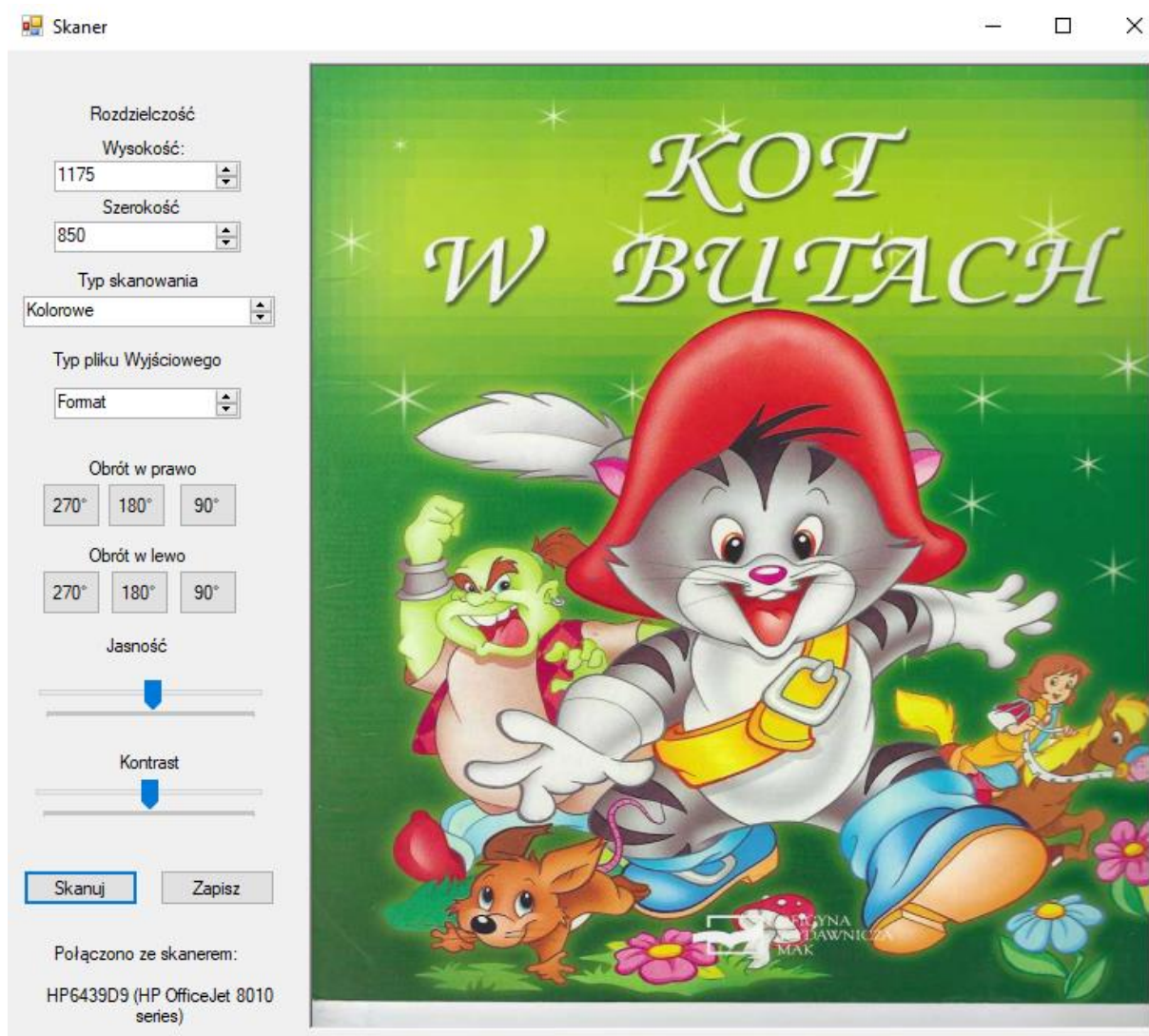
W momencie uruchomienia wyświetla nam się panel główny.



OBRAZ 12 – OBRAZ PANELU GŁÓWNEGO

W tym momencie powinniśmy wybrać rozdzielczość tryb skanowania i ustawić wstępnie wartość jasności i kontrastu jeżeli wiemy że dokument jest zbyt ciemny lub zbyt jasny.

Następnie naciskamy przycisk „Skanuj” powoduje to odczytanie ustawionych parametrów i wyświetlenie skanu.

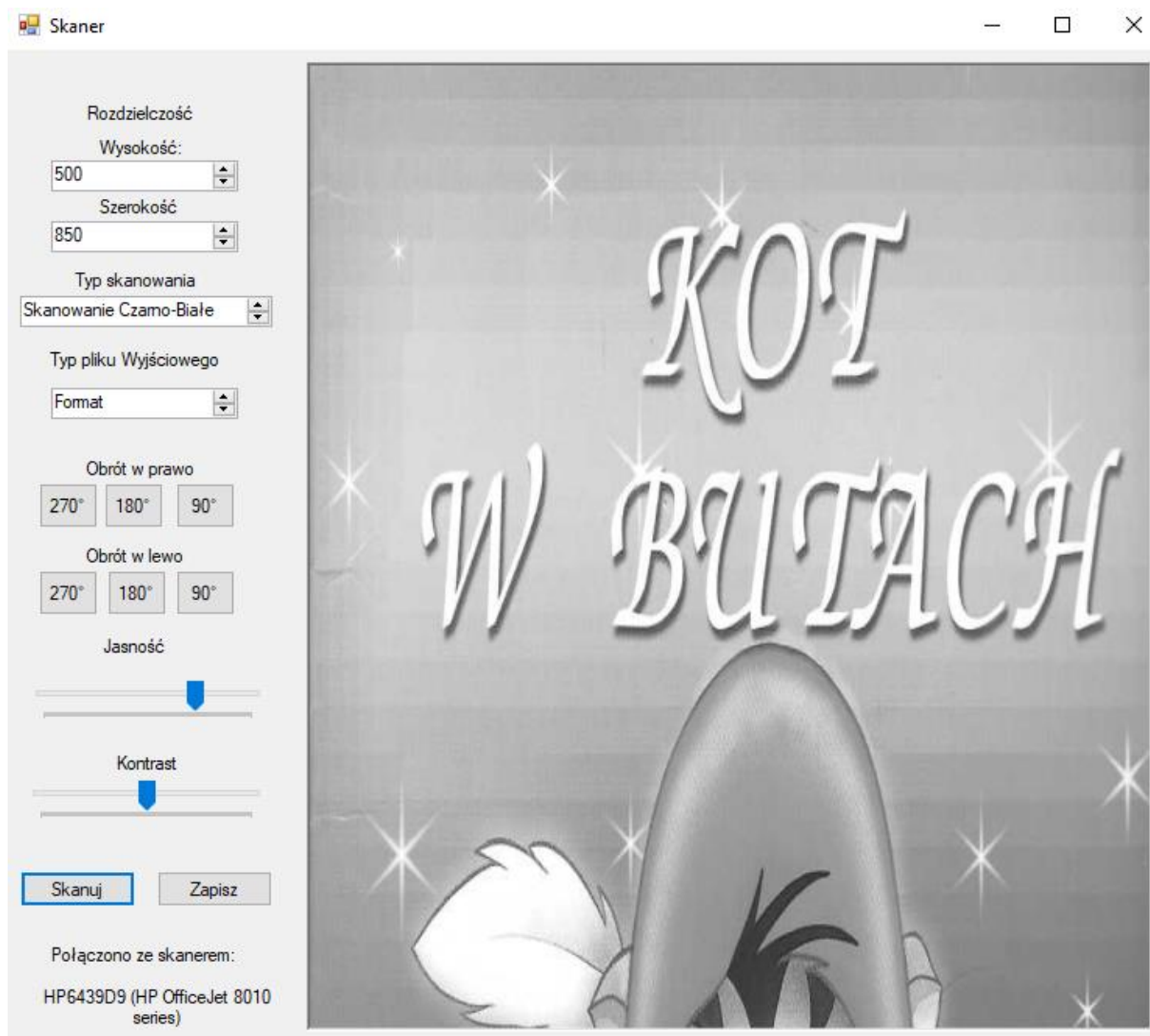


OBRAZ 13 – ZESKANOWANY OBRAZ

W tym momencie jeżeli wykryje jakikolwiek skaner pojawi się informacja z jakiego skanera korzysta i wyświetli zeskanowany dokument.

Teraz możemy korzystać z funkcji obrotu skanu.

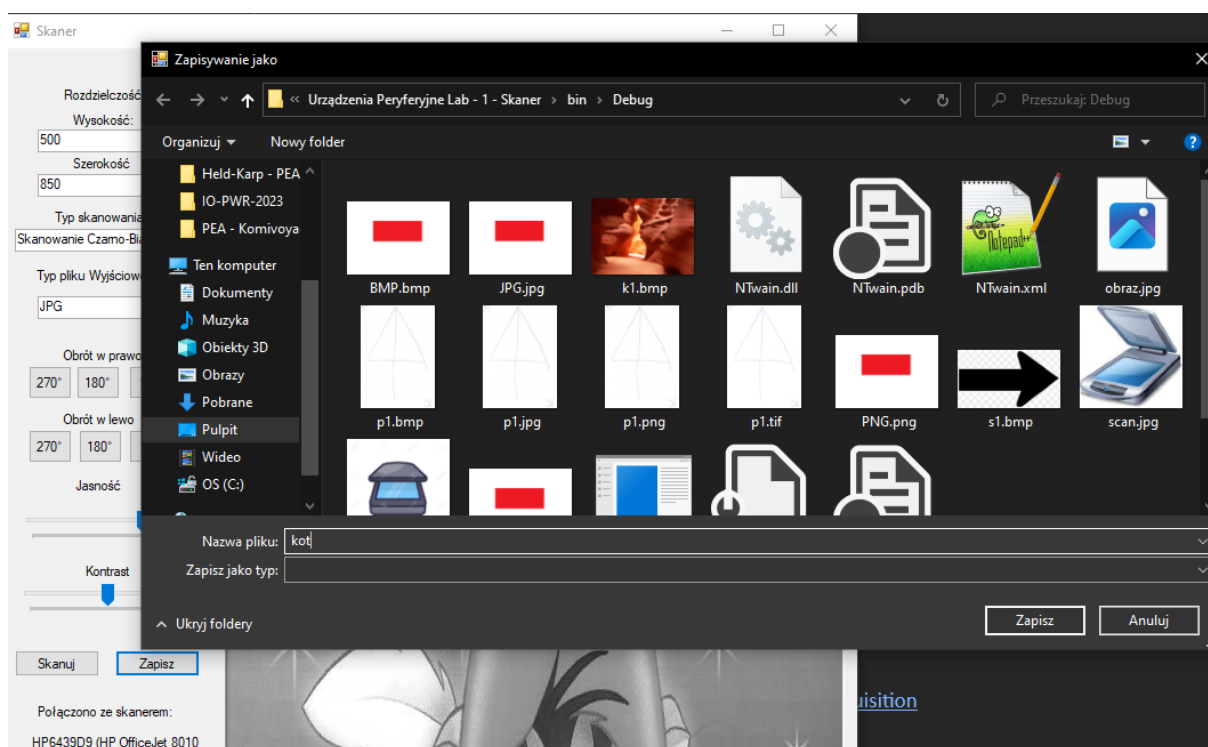
W przypadku gdybyśmy chcieli zmienić wielkość skanu, bądź parametry jasności, kontrastu lub tryb skanowania należy ponownie nacisnąć przycisk skanuj.



OBRAZ 14 – MODYFIKACJA USTAWIEŃ SKANERA

Jak możemy zauważyć skaner zeskanował obraz w formacie czarnobiałym z poniesioną jasnością i zmniejszoną wysokością.

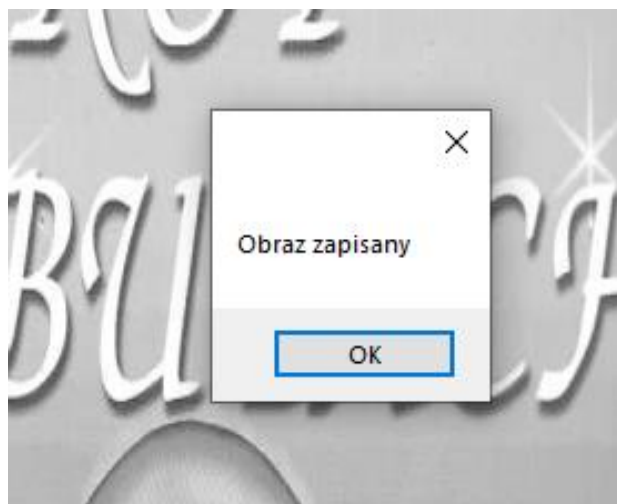
W tym momencie mając zeskanowany obraz możemy wybrać format zapisu i nacisnąć przycisk zapisz.



OBRAZ 15 – KOMUNIKAT O LOKALIZACJI ZAPISU

Pojawiło się okno wyboru lokalizacji i nazwy pliku wprowadzamy nazwę i wybieramy lokalizację.

Po naciśnięciu „Zapisz” potwierdzimy lokalizację a następnie skaner wykona ponownie skan i zapisze nasz plik w odpowiednim formacie.



OBRAZ 16 – KOMUNIKAT O POPRAWNYM ZAPISIE

Jeśli wszystko się powiedzie otrzymamy taki komunikat.

6) Wnioski

Jak możemy zauważyć korzystając z biblioteki WIA możemy w bardzo prosty sposób skanować dokumenty i równie prosto konfigurować parametry skanu. Największą trudnością w obsłudze tej biblioteki jest powiązanie ją z biblioteką odpowiedzialną za rysowanie. Jest to konieczne do bezproblemowego wyświetlenia obraz na ekranie oraz zapisania do pliku co wymaga strumieniowania danych.

Pomimo iż na zajęciach nie udało się uzyskać możliwości korekcji jasności i kontrastu to jednak dzięki wszechstronności i kompatybilności sterownika WIA bezproblemowo udało nam się połączyć z bezprzewodową domową drukarką z funkcją skaner i udoskonalić program. Dzięki temu końcowa wersja w pełni obsługuje skanowanie, zapis do pliku oraz korekcje jasności i kontrastu. Dodatkowo dodaliśmy komunikat o tym z jakim skanerem jesteśmy aktualnie połączeniu gdyż w przypadku podłączenia dwóch skanerów może to być bardzo istotna informacja.

W konfiguracji niedostępnej dla użytkownika ustawiono DPI na 100. Dzięki temu nie pojawiają się problemy ze zbyt dużym plikiem w stosunku wielkość obrazu do DPI co zapewnia wielkość pliku 1150x850 ustawiono także graniczne wartości kontrastu i jasności na poziomie od -1000 do 1000.

Dzięki takim limitom program zachowuje pełną sprawność niezależnie od użytkowania go przez użytkownika. W momencie nie wykrycia urządzenia program zwraca komunikat i taka sama sytuacja pojawia się w momencie braku trybu skanu bądź formatu zapisu. Dzięki czemu program jest wypełni zabezpieczony przed wszelkimi najbardziej prawdopodobnymi błędami powodującymi wysypanie się programu.

7) Referencje

Obrazy 1-3 - <https://miroslawzelent.pl/informatyka/skanery-ccd-cis-lide-ocr-twain/>

<https://hanzo.com.pl/Skanery-definicja-budowa-parametry>

https://pl.wikipedia.org/wiki/Skaner_p%C5%82aski

https://en.wikipedia.org/wiki/Windows_Image_Acquisition

<https://pl.wikipedia.org/wiki/JPEG>

https://pl.wikipedia.org/wiki/Portable_Network_Graphics

https://pl.wikipedia.org/wiki/Windows_Bitmap

https://pl.wikipedia.org/wiki/Tagged_Image_File_Format

https://www.123rf.com/photo_179015871_scanner-icon-in-vector-logotype.html