

URZĄDZENIA PERYFERYJNE

Skaner płaski

Autorzy:

Olga Tryk 263939

Dominika Nurek 260525

Grupa C

1. Wstęp

Skaner płaski służy do przekształcania obrazów lub tekstu na papierze na formę cyfrową. Skanuje on dokumenty lub zdjęcia, umieszczone na płaskiej powierzchni szklanej, za pomocą czujnika światła i soczewki. Czujniki (na przykład CCD - Charge-Coupled Device lub CIS - Contact Image Sensor) rejestrują odbite światło, rejestrując obrazy na powierzchni dokumentu. Obraz jest następnie przekazywany do komputera, gdzie może być zapisany, edytowany lub drukowany.

Na parametry skanera składają się między innymi:

- rozdzielczość (mierzona w punktach na cal (DPI - dots per inch), im większa rozdzielczość tym bardziej szczegółowy obraz oraz większy rozmiar pliku)
- kolory cyfrowe (określają, ile różnych kolorów może zostać zarejestrowanych przez skaner, zwykle obsługiwane są kolory RGB (czerwony, zielony, niebieski) lub CMYK (cyjan, magenta, żółć, czerń))
- balans bieli (pozwala dostosować skaner do różnych warunków oświetleniowych, aby uzyskać prawidłowe odwzorowanie kolorów, funkcja ta jest szczególnie istotna przy skanowaniu w warunkach zmieniającego się oświetlenia)
- de-mozaikowanie (mozaikowanie to efekt, który może wystąpić w przypadku niskiej rozdzielczości skanowania, gdzie brak jest wystarczającej ilości pikseli do precyzyjnego przedstawienia obrazu, funkcje de-mozaikowania pomagają poprawić tę sytuację)
- zoom (pozwala powiększać lub pomniejszać obraz podczas skanowania).

Istnieją różne biblioteki wspierające obsługę skanerów, takie jak TWAIN, WIA, SANE, ISIS. W naszym programie wykorzystana została biblioteka WIA. Dzięki niej mogliśmy uzyskać dostęp do różnych właściwości urządzenia, takich jak rozdzielczość, ustawienia kolorów itp., umożliwiając dostosowanie parametrów skanowania. Umożliwiła ona również przetworzenie i zapisanie obrazu w wybranym formacie (np. JPG, BMP, PNG).

2. Opis programu

Program został napisany w języku C#. Daje on możliwość zeskanowania obrazu oraz zmiany niektórych parametrów. Jest możliwość wybrania jednej z trzech opcji kolorystycznych obrazu (kolorowy, czarno-biały, szary), ustawienia DPI, a także można ustawić jaki procent obrazu zostanie zeskanowany.

Modyfikowane parametry identyfikowane są za pomocą odpowiadających im stałych liczbowych zdefiniowanych w bibliotece WIA. Aby zmienić parametr wystarczy ustawić nową wartość. W przypadku zmiany koloru biblioteka WIA posiada zdefiniowane wartości flag parametru WIA_IPS_CUR_INTENT (6146).

Uwaga: Aby móc skorzystać z biblioteki WIA w środowisku Visual Studio należało dodać odwołanie "Microsoft Windows Image Acquisition Library v.2.0" i upewnić się, że parametr "Osadź typy międzyoperacyjne" ma wartość "False".

Kod programu:

```
C/C++
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using WIA;

namespace Skaner
{
    public partial class Form1 : Form
    {
        private Device device;
        public Form1()
        {
            InitializeComponent();
            Connect();

            cmbColorType.Items.Add("Kolor");
            cmbColorType.Items.Add("Szary");
            cmbColorType.Items.Add("Czarno-Biały");
            cmbColorType.Text = "Kolor";

            cmbSize.Items.Add("100%");
            cmbSize.Items.Add("50%");
            cmbSize.Items.Add("25%");
            cmbSize.Text = "100%";
        }

        private void Connect()
        {
            DeviceManager manager = new DeviceManager();

            foreach (DeviceInfo deviceInfo in manager.DeviceInfos)
            {
                if (deviceInfo.Type == WiaDeviceType.ScannerDeviceType)
                {
                    device = deviceInfo.Connect();
                    break;
                }
            }

            if(device == null)
            {
                MessageBox.Show("Nie znaleziono skanera.");
            }
        }
    }
}
```

```

private void btnScan_Click(object sender, EventArgs e)
{
    //Czy wartość dpi jest poprawna
    int dpi;
    if(!int.TryParse(txtDPI.Text, out dpi))
    {
        MessageBox.Show("Podana wartość DPI nie jest liczbą.");
        return;
    }
    if(dpi < 50 || dpi > 600)
    {
        MessageBox.Show("Podana wartość DPI znajduje się poza dozwolonym
zakresem (50 - 600).");
        return;
    }

    var scannerItem = device.Items[1];

    scannerItem.Properties["6147"].set_Value(dpi); //horizontal resolution
    scannerItem.Properties["6148"].set_Value(dpi); //vertical resolution

    float fraq = 1;
    if(cmbSize.Text.Equals("50%"))
    {
        fraq = 0.5f;
        pictureBox1.Size = new Size(633, 440);
    }
    else if(cmbSize.Text.Equals("25%"))
    {
        fraq = 0.25f;
        pictureBox1.Size = new Size(633, 220);
    }

    scannerItem.Properties["6151"].set_Value(8.3f * dpi); //horizontal
extent
    scannerItem.Properties["6152"].set_Value(11.6f * dpi * fraq);
//vertical extent

//6146 - current intent
    if (cmbColorType.Text.Equals("Kolor"))
    {
        scannerItem.Properties["6146"].set_Value(1);
    }
    else if(cmbColorType.Text.Equals("Szary"))
    {
        scannerItem.Properties["6146"].set_Value(2);
    }
    else if(cmbColorType.Text.Equals("Czarno-Biały"))
    {
        scannerItem.Properties["6146"].set_Value(4);
    }
}

```

```

else
{
    scannerItem.Properties["6146"].set_Value(1);
}

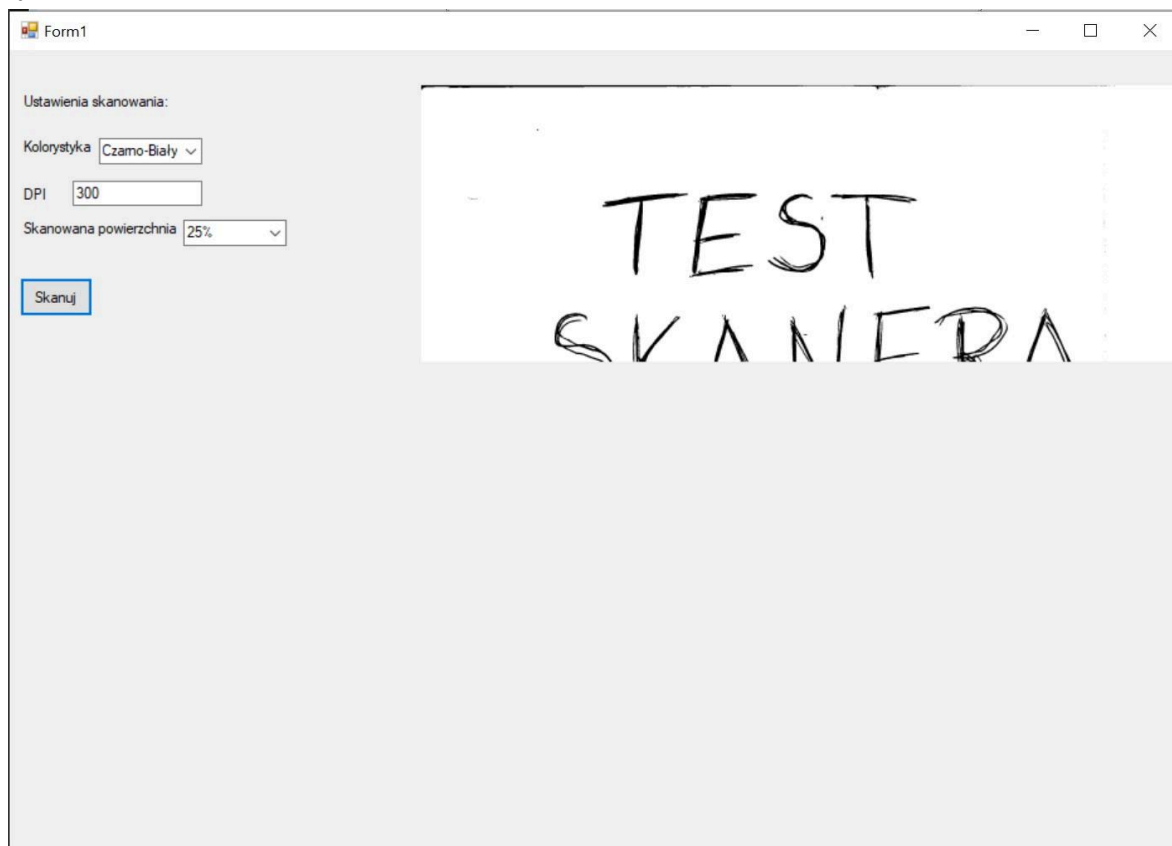
ImageFile imageFile =
(ImageFile)scannerItem.Transfer(WIA.FormatID.wiaFormatJPEG);
string path = "C:\\Users\\lab\\Desktop\\image.jpg";

if (File.Exists(path))
{
    File.Delete(path);
}
imageFile.SaveFile(path);

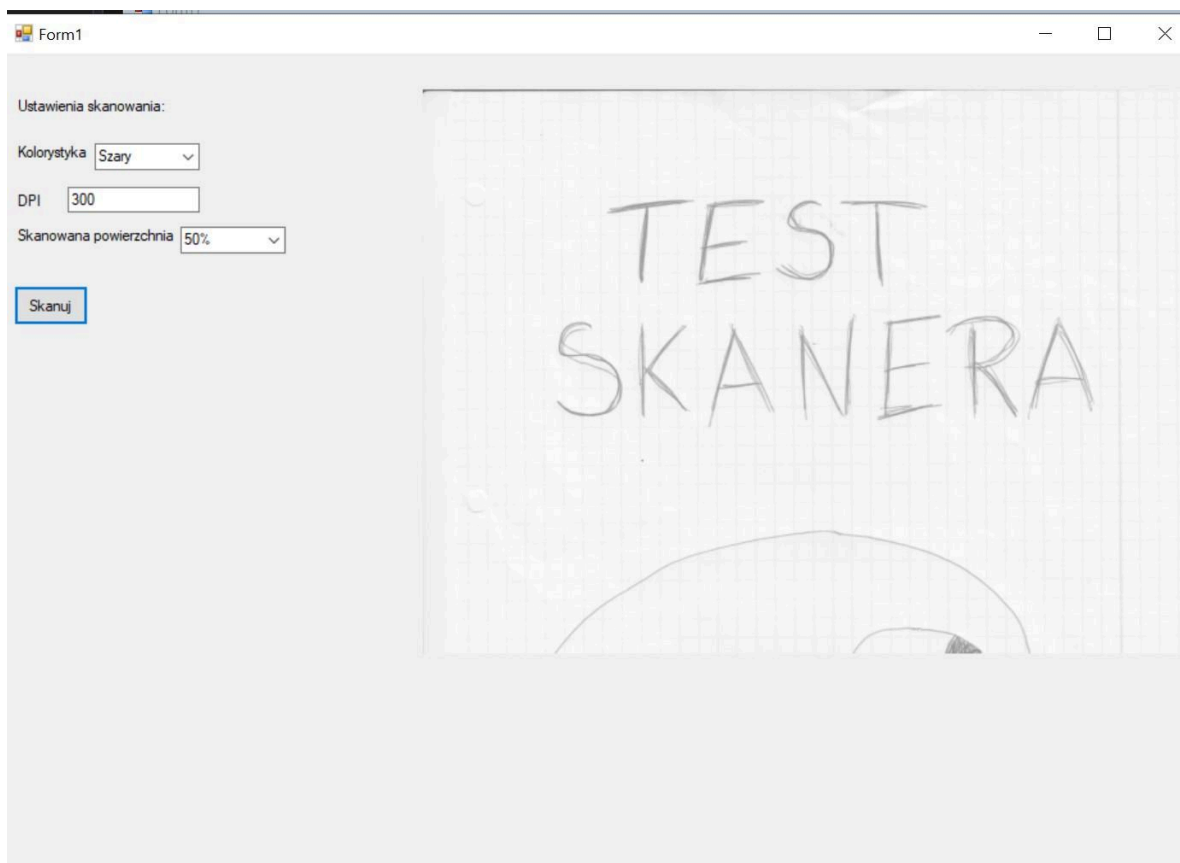
pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;
pictureBox1.ImageLocation = path;
}
}
}

```

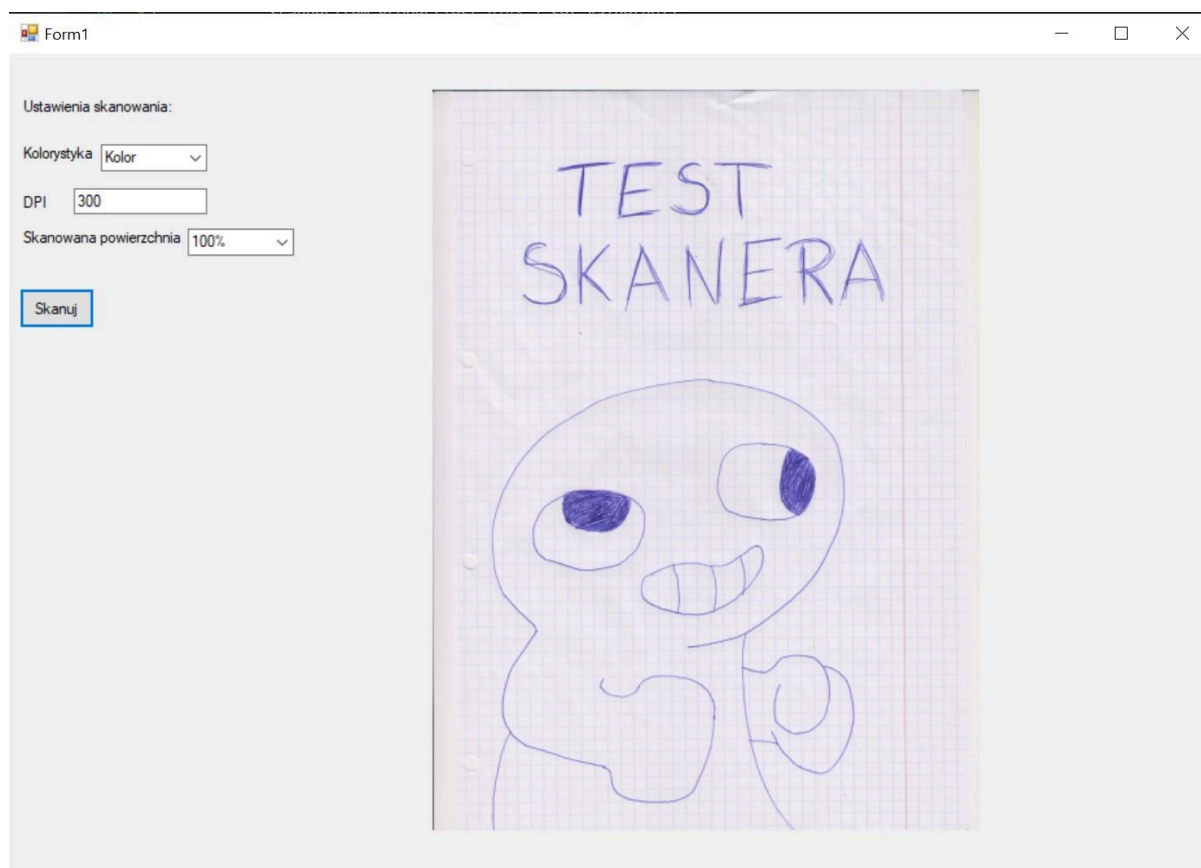
Zrzuty ekranu:



Rys. 1 Okno programu, kolorystyka czarno-biała, 300 DPI, skan 25% powierzchni

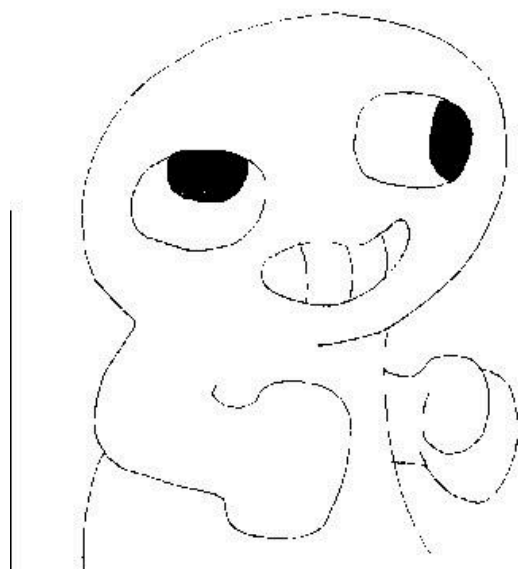


Rys. 2 Okno programu, kolorystyka szara, 300 DPI, skan 50% powierzchni



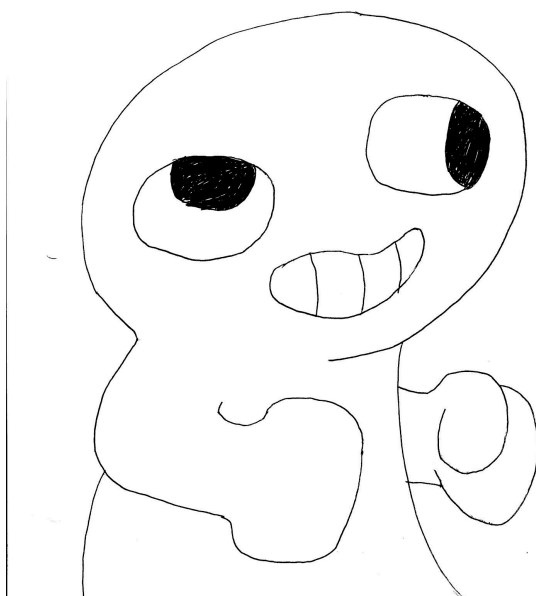
Rys. 3 Okno programu, kolorystyka kolorowa, 300 DPI, skan 100% powierzchni

TEST
SKANERA



Rys. 4 Skan zapisany w formacie jpg, kolorystyka czarno - biała, 50 DPI, 100% powierzchni

TEST
SKANERA



Rys. 5 Skan zapisany w formacie jpg, kolorystyka czarno - biała, 600 DPI, 100% powierzchni

3. Wnioski

Podstawą tworzenia programu było korzystanie z dokumentacji biblioteki WIA. Trudnością była identyfikacja parametrów, które należy zmienić, aby otrzymać pożądany efekt.

Przykładowo, przy zmianie rozdzielczości należy oddzielnie ustawić rozdzielczość pionową i poziomą. W przypadku niezaimplementowanej opcji zmiany orientacji obrazu trudność sprawiło znalezienie definicji dozwolonych wartości edytowanego parametru.

Kod zmiany orientacji byłby bardzo podobny do zmiany kolorystyki. Edytowany byłby parametr WIA_IPS_ORIENTATION (6156) lub WIA_IPS_ROTATION (6157). Parametr orientation definiuje położenie skanowanego pliku względem kierunku skanowania, a parametr rotation zmienia położenie obrazu po zeskanowaniu.

Dla obu parametrów zdefiniowane są wartości:

0 - orientacja pionowa, domyślna wartość

1 - orientacja pozioma, obrót o 90 stopni

2 - obrót o 180 stopni

3 - obrót o 270 stopni.