

# Spis treści

<b>Wstęp</b>	<b>3</b>
<b>Cel i zakres pracy</b>	<b>4</b>
<b>1 Część teoretyczna</b>	<b>5</b>
1.1 Rys historyczny . . . . .	5
1.1.1 Wywoływanie zdjęć . . . . .	5
1.1.2 Transmisja obrazów . . . . .	6
1.1.3 Cyfrowe przetwarzanie obrazów . . . . .	6
1.2 Przegląd istniejących rozwiązań . . . . .	7
1.2.1 Photoshop . . . . .	7
1.2.2 ImageJ . . . . .	8
1.2.3 Fiji . . . . .	9
1.2.4 GIMP . . . . .	10
1.2.5 Adobe Substance Designer . . . . .	11
<b>2 Zakres użytych technologii i opis wykorzystywanych narzędzi</b>	<b>12</b>
2.1 OpenCV . . . . .	12
2.2 Visual Studio Code . . . . .	12
2.3 Git . . . . .	12
2.4 GitHub . . . . .	12
<b>3 Realizacja projektu</b>	<b>13</b>
3.1 Zastosowania . . . . .	13
3.2 Wymagania funkcjonalne i niefunkcjonalne . . . . .	13
3.2.1 Wymagania funkcjonalne . . . . .	13

3.2.2 Wymagania niefunkcjonalne . . . . .	13
<b>4 Podsumowanie</b>	<b>14</b>
<b>Bibliografia</b>	<b>15</b>
<b>Spis rysunków</b>	<b>18</b>
<b>Spis tabel</b>	<b>19</b>

# Wstęp

Przetwarzanie obrazów to zbiór wielu algorytmów które należy rozróżnić na dwie dziedziny ze względu na rodzaj reprezentowanych i manipulowanych informacji. Pierwsza dotyczy analogowej reprezentacji danych [1], gdzie informacje są w formie ciągłych sygnałów odwzorowujących obrazy. Operacje na nich są realizowane przez modyfikacja impulsów elektrycznych reprezentujących informacje. Technologia ta umożliwiła rozwój transmisji telewizyjnej [2]. Cyfrowe przetwarzanie obrazów [3] jest nowszą technologią. Dane są reprezentowane poprzez wielowymiarowe macierze wartości [4]. Mogą one odpowiadać kolorowi obrazu w danym miejscu, wartościom chmury punktów lub innym informacjom które mogą zostać przetworzone cyfrowo. Modyfikacja odbywa się przez zmiany wartości macierzy. Obie dziedziny mają swoje miejsca i zastosowania ale niniejsza praca zajmie się cyfrowym przetwarzaniem obrazów.

Przetwarzania obrazów jest wykorzystywane w wielu różnych dziedzinach:

- **Medycyna:** Diagnostyka, planowanie zabiegów chirurgicznych i monitorowanie stanu pacjentów [5]. Dane medyczne uzyskiwane w formie obrazów cyfrowych są ustandaryzowane przez Digital Imaging and Communications in Medicine [6].
- **Przemysł:** Kontrola jakości, automatyzacja i inspekcja. Systemy wizyjne potrafią szybko i dokładniej odkryć defekty produktów na liniach produkcyjnych [7].
- **Bezpieczeństwo:** Monitorowanie i rozpoznawanie twarzy. Drony ze wsparciem przetwarzania obrazów są używane przez służby bezpieczeństwa w trudnych warunkach [8].
- **Rozrywka:** Tworzenie efektów specjalnych dla filmów, gier i animacji. Transformacje przestrzeni kolorów i korekcie nagrań w celu poprawienia efektu końcowego [9].

Przetwarzanie obrazów jako manipulacja danymi cyfrowymi została stworzona przez programistów [10]. Kod to podstawowa wersja interakcji człowieka z komputerem, ale większość użytkowników tych urządzeń nie potrafi programować. Żeby dziedzina tej nauki była przystępna dla większej liczby osób należy stworzyć aplikację z prostym i funkcjonalnym graficznym interfejsem użytkownika.

W tej pracy naukowej proponujemy nowe oprogramowanie do przetwarzania obrazów oparte na bibliotece OpenCV. Aplikacja posiada funkcjonalny interfejs graficzny, który ułatwia użytkownikom końcowym obsługę tej biblioteki bez potrzeby znajomości programowania.

# Cel i zakres pracy

Celem tej pracy jest zaprojektowanie i implementacja oprogramowania do przetwarzania obrazów z funkcjonalnym interfejsem graficznym.

Zakres pracy objął analizę istniejących rozwiązań na podstawie której stwierdzono, że nie ma aktualnego i gotowego produktu który realizuje dokładnie założenie zastąpienia programistycznego podejścia do obsługi bibliotek przetwarzania obrazów. Jest wiele rozwiązań które realizuje część biblioteki lub jej obsługa nie pozwala na przetwarzanie obrazów w niedestruktacyjny sposób bez dodatkowego wysiłku od użytkownika jak używanie warstw czy inkrementalnych zapisów.

Jako platformę wybrano komputer osobisty z powodu znaczącej mocy obliczeniowej, dostępu do dużej ilości pamięci RAM oraz do operacji procesorów x86 z których biblioteki do przetwarzania obrazów czasem korzystają do akceleracji obliczeń [11]. Jako docelowy system operacyjny wybrano Microsoft Windows 11 ponieważ jest to system używany przez autora tej pracy. Do tworzenia aplikacji skorzystano z języka C# oraz Windows Presentation Framework.

W ramach czasu, który został poświecony na niniejszy projekt zakres możliwości ograniczono do prostych operacji przetwarzania obrazów bez uwzględniania wideo oraz wszelkiego rodzaju danych oprócz samych obrazów.

Pomysł na ten projekt powstał dzięki doświadczeniu autora pracy w używaniu różnych systemów opartych o edytory typu węzlowego jak Adobe Substance Designer, edytor Blueprints w Unreal Engine oraz edytor shaderów czy geometry nodes w Blenderze. Ten typ interfejsu pozwala na interakcję użytkownika z wizualnym przedstawieniem jakichś metod, funkcji czy bloków logicznych pozwalających na przesyłanie wartości w podobny sposób jak można robić to w kodzie, ale zdecydowanie bardziej przystępne dla osób które nie mają odpowiedniego doświadczenia w programowaniu.

Dużą zaletą tego podejścia w porównaniu do edycji w programach posługujących się warstwami jest to, że możemy wrócić w każdym momencie do dowolnej poprzedniej operacji. Użytkownik może zmienić wartość operacji, zmienić ich kolejność i w żadnym momencie nie traci wyniku operacji, ponieważ każda zmiana powoduje obliczenie rezultatu jeszcze raz. Przeliczenie całego drzewa na nowym obrazie wejściowym i otrzymanie wyniku którego się spodziewamy to zmiana jednego parametru.

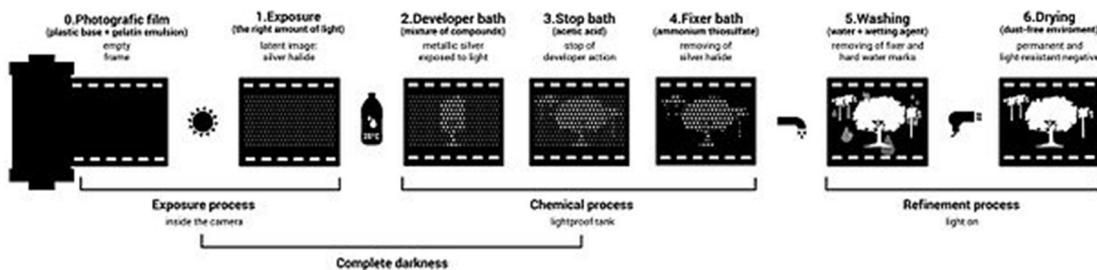
# 1 Część teoretyczna

## 1.1 Rys historyczny

Dziedzina tej pracy jaką jest przetwarzanie obrazów ma początki swojej historii wcześniejszej niż urządzenia które obecnie nazywamy komputerami.

### 1.1.1 Wywoływanie zdjęć

Przetwarzanie obrazów to nie jest technologia, która mogła zacząć istnieć po wynalezieniu komputerów. Używając aparatów korzystających z kliszy filmowej, po ekspozycji należy go poddać procesowi wywoływania Rys. 1. Polega on na wyciągnięciu pierwotnego efektu naświetlenia do obrazu, który oddaje scenę uchwyconą przez aparat [12]. Proces w przypadku zdjęć aparatem na kliszę polega na zanurzaniu jej w odpowiednich związkach chemicznych na określone ilości czasu by otrzymać zamierzony efekt.



Rys. 1: Wywoływanie biało czarnej kliszy [13].

Istnieją wariacje na temat takiego przetwarzania, różni się ono trochę w zależności od technologii kliszy. W niektórych przypadkach należy wywołać pozytyw zamiast negatywu [14]. Następnie po wywoływaniu można poddać obróbce dalszymi chemikaliami jak np. siarczek sodu dla uzyskania efektu sepii [15].

### **1.1.2 Transmisja obrazów**

Technologia którą znamy dobrze w obecnych czasach powstała w latach 20. XX wieku dzięki pracy John Logie Baird'a [2]. Zaprezentował on pierwszą na świecie transmisję telewizyjną na żywo. W 1928 roku [16] firma założona przez niego emitowała też pierwsze nagranie przez atlantyk. Wszystko to dzięki przetwarzaniu analogowego sygnału video w celu zmniejszenia ilości danych potrzebnych do wyświetlenia obrazu w telewizji [17].

### **1.1.3 Cyfrowe przetwarzanie obrazów**

Początki nowoczesnego przetwarzania obrazów zostały stworzone w latach 60. XX w. w Bell Laboratories, Jet Propulsion Laboratory, Massachusetts Institute of Technology i University of Maryland [10]. Początkowymi obszarami zastosowania były obrazy satelitarne, przesyłanie obrazów przez linie telefoniczne, diagnostyka obrazowa, videofony, rozpoznawanie znaków i ulepszanie fotografii.

Inicjalnie dużo skupiano się na ulepszeniu jakości obrazu. Pierwszym znaczącym użyciem tych technologii było mapowanie powierzchni księżyca za pomocą zdjęć z sondy kosmicznej w 1964 roku, gdzie naukowcy z Jet Propulsion Laboratory na podstawie tysięcy zdjęć odtworzyli powierzchnię księżyca. Przy następnej okazji mieli dostęp do 100000 zdjęć, na podstawie których mogli stworzyć mapę topograficzną, mapę kolorową oraz panoramiczną mozaikę księżyca, które przyczyniły się do pierwszego lądowania człowieka na księżycu [3].

Technologia ta jednak była ograniczona przez bardzo małe możliwości oraz trudną dostępność komputerów tamtych czasów. Wraz z ich rozwojem i wzrostem popularności cyfrowe przetwarzanie obrazów przestało być trudno dostępną dziedziną naukową i stopniowo zostało wdrażane do naszego codziennego życia. Obecnie około 80% populacji krajów rozwiniętych jest w posiadaniu smartfona [18], który na swoim wyposażeniu ma kamerę cyfrową. Są one ograniczone poprzez fizyczny rozmiar matrycy oraz jakość soczewek, które mogą zmieścić się w telefonach komórkowych. Pomimo tego dzięki przetwarzaniu obrazów można uchwycić za ich pomocą imponujące zdjęcia [19]. Nowoczesne telefony są w stanie zachować na zdjęciu nocnym detale które może być ciężko zobaczyć ludzkim okiem [20] przez szybkie łączenie wielu obrazów o różnych długościach ekspozycji, redukcji szumów i korekcji kolorów.

## 1.2 Przegląd istniejących rozwiązań

### 1.2.1 Photoshop



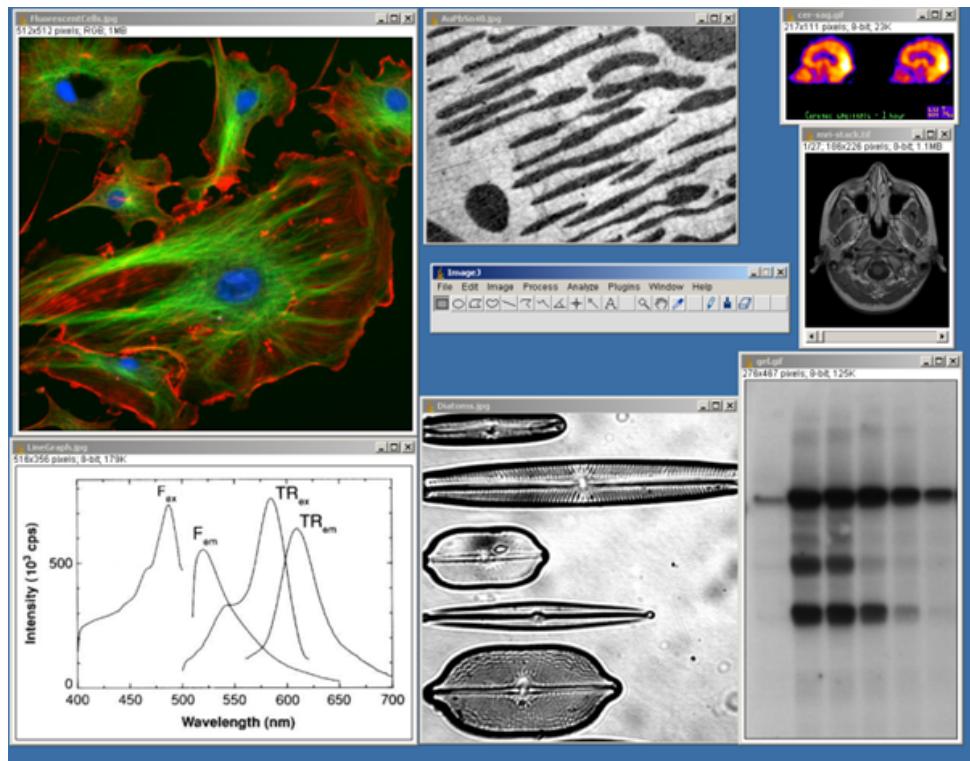
Rys. 2: Adobe Photoshop [21].

Wydany w 1990 roku do tworzenia i edytowania obrazów rastrowych jest jednym z najbardziej popularnych programów, a jego nazwa przedeńała się do języka potocznego jako zamiennik dla fotomontażu. Użytkownikami tego oprogramowania są przeważnie artyści, fotografowie i twórcy internetowych memów. To narzędzie jest przystosowane do łatwości obsługi przez mniej zaawansowanych użytkowników i wszystko posiada przyjazny graficzny interfejs użytkownika.

Funkcje, które pozwalają na przetwarzanie obrazów często nie odnoszą się bezpośrednio do operacji zawartych w znanych bibliotekach do przetwarzania obrazów. Ich obsługa jest uproszczona i przystosowana do bardziej artystycznych zastosowań. Bez zastosowania szczególnej ostrożności - jak tworzenie nowych warstw po każdej operacji lub częste zapisywanie kopii zapasowej obrazu – przetwarzanie obrazów często jest destruktywne, nie możemy odnieść się do dowolnego momentu w ciągu naszych operacji w celu dopasowania jej parametrów. Przetwarzanie innego obrazu za pomocą tego samego procesu jest problematyczne, gdyż wymaga to

ciągłego śledzenia warstw. Wiele operacji trzeba powtórzyć indywidualnie ponieważ nie zapisujemy jej parametrów tylko rezultat poprzedniego wykonania. Wiele bardziej zaawansowanych procesów nie jest możliwych do zrealizowania za pomocą standardowych opcji dostępnych w programie. Historia operacji jest też ograniczona i nie można wyświetlić pełnej historii na jednym ekranie - można jedynie zmieniać stan aktualny na wcześniejszy.

### 1.2.2 ImageJ



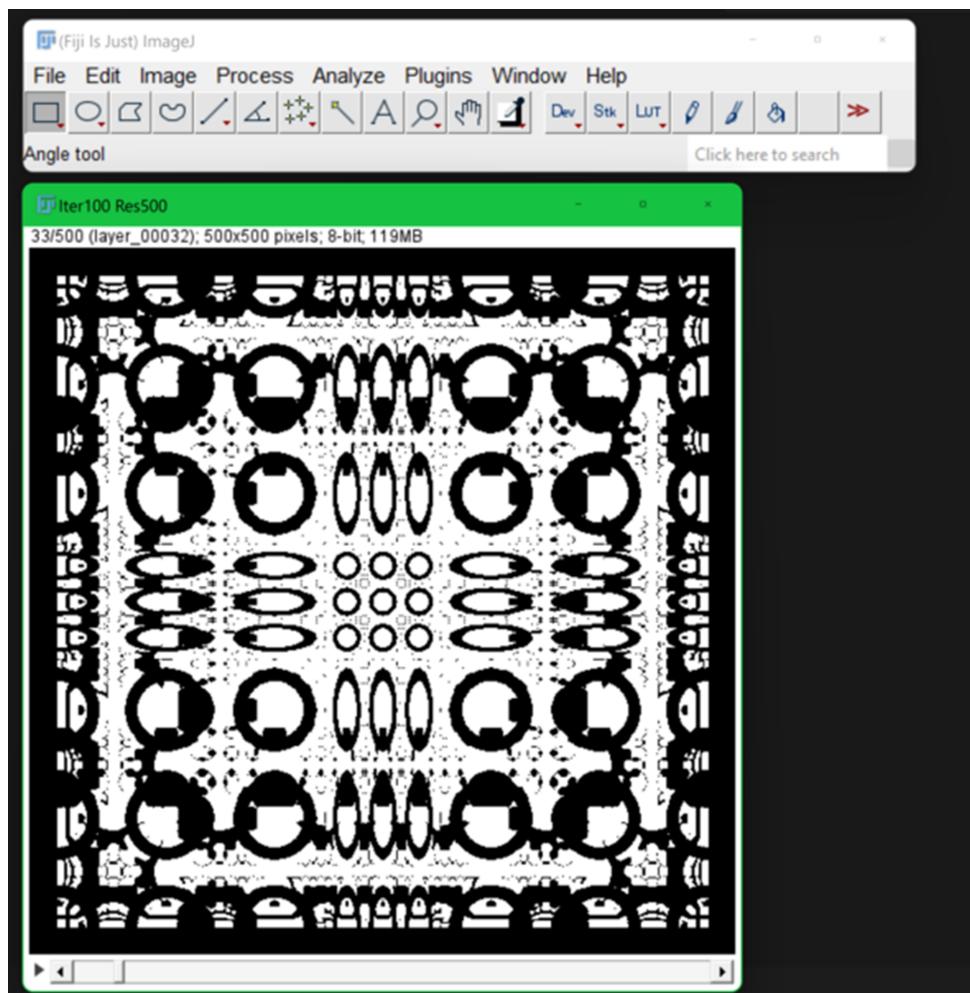
Rys. 3: ImageJ [22].

Wydany w 1997 roku program ImageJ został wykonany przez National Institutes of Health, został tworzony podstawowo z zamiarem analizy obrazów w zastosowaniach medycznych. Odtwarza dokładnie wiele operacji z bibliotek do przetwarzania obrazów i kod aplikacji jest otwartoźródłowy - każdy może go zobaczyć, pomóc w rozwoju lub stworzyć własną wersję. Napisany został w języku Java i ma bardzo niskie wymagania sprzętowe jak na obecne czasy.

Program jest jednak przestarzały, mogą pojawić się problemy z uruchomieniem na nowszzych systemach. Interfejs użytkownika wygląda archaicznie oraz jego układ jest niepraktyczny biorąc pod uwagę przetwarzanie obrazów. Wszystkie operacje są schowane pod 1-2 poziomami

menu. Operacje są destrukcyjne - zmieniają dane, które przetwarzamy więc użytkownik musi pilnować aby mieć kopię swoich obrazów. Nie da się przygotować ciągów operacji wcześniej za pomocą interfejsu graficznego, trzeba użyć do tego ich języka programowania ImageJ Macro [23]. Od dawna nie jest rozwijany, został zastąpiony przez ImageJ2, który pozwala na przetwarzanie obrazów wielowymiarowych z dodatkowymi danymi np. z mikroskopów elektronowych, skanerów itp.

### 1.2.3 Fiji



Rys. 4: Fiji - sukcesor ImageJ [24].

Projekt open source oparty o ImageJ2. Oprócz podstawowych operacji wbudowanych w ImageJ posiada też wiele pluginów znacząco rozszerzających możliwości programu. Są one skupione na wspomaganiu przetwarzania obrazów skupionych na dziedzinie neurobiologii, ale możliwości są na tyle szerokie, że wiele innych dziedzin nauki z niego korzysta.

Podstawowe działanie programu nie różni się od ImageJ więc, wszystkie jego problemy są też tutaj obecne. Jednakże poprawiona została jego kompatybilność z najnowszymi systemami operacyjnymi.

#### 1.2.4 GIMP

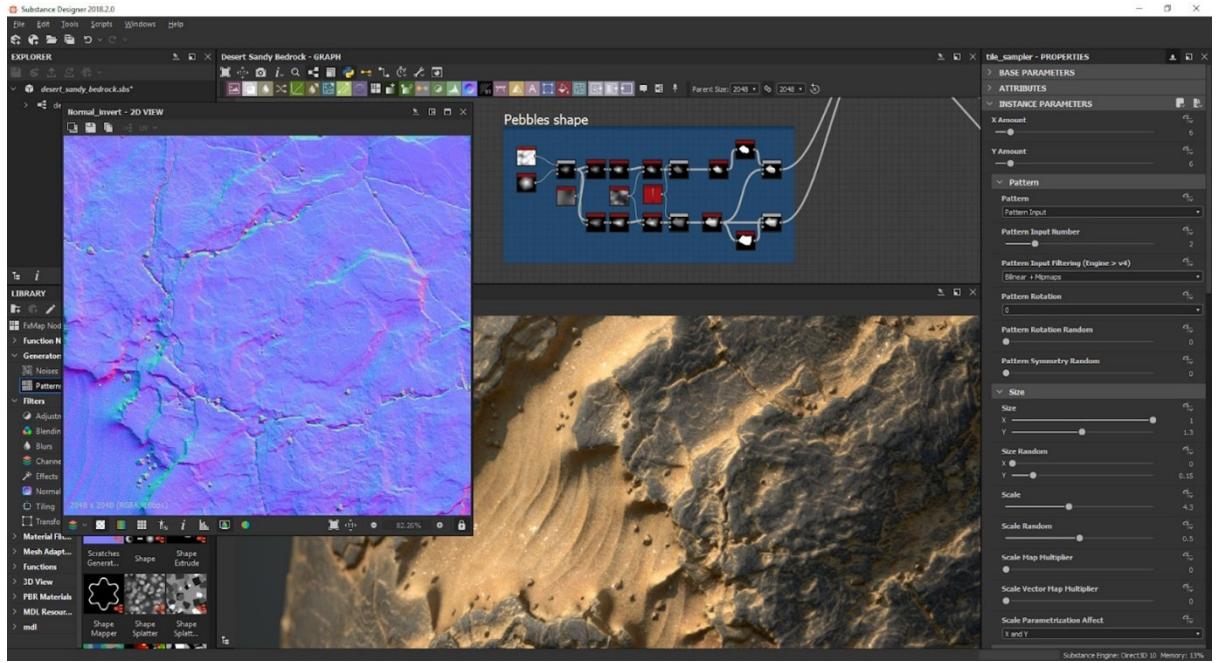


Rys. 5: GIMP wersja 2.10 [25].

Wydany w 1998 roku GNU Image Manipulation Program jest tworzony jako darmowa konkurencja dla Photoshopa, którego licencja jest miesięczną subskrypcją. Program też w pierwszej kolejności jest tworzony z myślą o artystach, ale posiada więcej zaawansowanych opcji jak np. konwolucje macierzowe które można dowolnie edytować, dając duże możliwości filtrowania obrazów.

Pomimo odwzorowaniu większej liczby operacji ze standardowych bibliotek do przetwarzania obrazów i większej kontroli nad niektórymi z nich nadal występuje problem destruktywnego przetwarzania obrazów. Wynika to z pracy bezpośrednio na obrazie jak w Photoshopie i wymaga szczególnej uwagi przy zarządzaniu warstwami aby nie tracić bezpowrotnie swoich pośrednich operacji w dłuższym ciągu.

### 1.2.5 Adobe Substance Designer



Rys. 6: Adobe Substance Designer [26].

Wydany w 2007 roku program był główną inspiracją dla tego projektu. Jego dużą zaletą jest to, że tworzenie algorytmów bazuje na układaniu bloczków (nodów). Ułożony graf jest wykonywany na obrazie, który importujemy do programu lub generujemy go od początku w nim. Każdy dowolny node można kliknąć i zmienić wszystkie parametry jego operacji, niezależnie w którym miejscu ciągu znajduje się. Jego wynik i wszystkie następne operacje, które są zależne od niego są obliczane ponownie na podstawie zmienionego wyniku. Zmiana przetwarzanego obrazu polega na przeciągnięciu połączenia z obecnego węzła z naszym plikiem wejściowym na nowy bloczek. Następnie propagowana jest zmiana na wszystkie kolejne operacje, których wynik jest aktualizowany automatycznie.

Opisywany program ten ma wiele ograniczeń związanych z tym, że nie jest stworzony do ogólnego przetwarzania obrazów. Został natomiast stworzony do tworzenia materiałów/tekstur do grafiki komputerowej. Obrazy są ograniczone do boków o długości  $2^n$ , najlepiej kwadratowych. Parametry operacji są często uproszczone, ponieważ mimo większej złożoności aplikacji, jest ona nadal skierowana do artystów. Rodzaje operacji i wspierane formaty pikseli w pliku są przystosowane do wymagań grafiki komputerowej.

## **2 Zakres użytych technologii i opis wykorzystywanych narzędzi**

Zastosowane technologie:

- **Otwarty kod źródłowy:** OpenSource jest spoko

### **2.1 OpenCV**

Jedną z popularnych bibliotek do przetwarzania obrazów jest OpenCV. OpenCV to darmowy, otwartoźródłowy pakiet oprogramowania dla systemów operacyjnych Windows, macOS i Linux. Zapewnia szeroki zakres funkcji do przetwarzania obrazów, w tym: podstawowe operacje takie jak filtrowanie, skalowanie i konwersja formatów, wykrywania i rozpoznawania obiektów czy obsługę algorytmów do analizy wideo.

### **2.2 Visual Studio Code**

### **2.3 Git**

### **2.4 GitHub**

## 3 Realizacja projektu

### 3.1 Zastosowania

- **Tworzenie gotowych ciągów operacji:** Czasem potrzebne jest przetworzenie małej ilości obrazów, lub problem jest na tyle mało skomplikowany, że nie ma sensu szukać programisty, który stworzy dla nas program, ale bez wiedzy o programowaniu zrobienie tego w kodzie samodzielnie może być zbyt ciężkie. Program z interfejsem graficznym pozwoli na stworzenie procesów nawet przez osoby mniej techniczne.
- **Uczenie się:** Osoby chcące poznać techniki przetwarzania obrazów będą mogły w prosty sposób bez znajomości programowania zobaczyć na własne oczy działanie funkcji, ich interakcję ze sobą oraz łatwo dopasowywać ich parametry.
- **Komunikacja:** Dzięki temu oprogramowaniu programista może łatwiej i efektywniej porozumieć się z osobami mniej zaznajomionymi z programowaniem i dużo szybciej wprowadzać zmiany w porównaniu z pisaniem kodu i jego kompilacją przy każdej zmianie.

### 3.2 Wymagania funkcjonalne i niefunkcjonalne

Oprogramowanie opisane w tej pracy to **NoodleCV**. Następne podrozdziały przedstawią wymagania z jakimi trzeba było się zmierzyć w trakcie tworzenia aplikacji.

#### 3.2.1 Wymagania funkcjonalne

Wymagania funkcjonalne?

- Przetwarzanie obrazów

#### 3.2.2 Wymagania niefunkcjonalne

Niefunkcjonalne wymagania?

- Łatwość użytkowania

## **4 Podsumowanie**

*Na koniec należy podsumować, co było celem pracy. Czy cel ten został osiągnięty. Jakie są możliwe wnioski. Czy są dalsze możliwości rozwoju.*

## Bibliografia

1. Chakravorty, P. What Is a Signal? [Lecture Notes]. *IEEE Signal Processing Magazine* **35**, 175–177. <https://api.semanticscholar.org/CorpusID:52164353> (2018).
2. The Televisor: Successful Test of New Apparatus. *The Times (London)*, 9 column C. <https://www.bairdtelevision.com/the-televisor-successful-test-of-new-apparatus-1926.html> (1926).
3. Gonzalez, R. C. & Woods, R. E. *Digital image processing* 3rd, 23–28. ISBN: 978-0-13-168728-8 (Prentice Hall, Upper Saddle River, N.J., 2008).
4. *OpenCV: cv::Mat Class Reference* OpenCV. [https://docs.opencv.org/3.4/d3/d63/classcv\\_1\\_1Mat.html#details](https://docs.opencv.org/3.4/d3/d63/classcv_1_1Mat.html#details).
5. Szymon, K. *Przetwarzanie obrazów* <https://skszymon.eu/blog/2023/cv-przetw-obrazow/>.
6. *DICOM Part 1: Introduction and Overview* National Electrical Manufacturers Association. [https://dicom.nema.org/medical/dicom/current/output/chtml/part01/chapter\\_1.html#sect\\_1.1](https://dicom.nema.org/medical/dicom/current/output/chtml/part01/chapter_1.html#sect_1.1).
7. Bokhan, K. *Computer Vision in Manufacturing: A Guide to Integration* N-iX. <https://www.n-ix.com/computer-vision-manufacturing/>.
8. Enterprise, D. *How Can Police, Firefighters and Search and Rescue Professionals Use Drones to Keep the Public Safe?* <https://enterprise-insights.dji.com/blog/droneshelp-how-can-police-firefighters-and-search-and-rescue-professionals-use-drones-to-keep-the-public-safe>.
9. Bertalmio, M. *Image Processing for Cinema* ISBN: 9780429067501 (lut. 2014).
10. Rosenfeld, A. Picture Processing by Computer. *ACM Comput. Surv.* **1**, 147–176. ISSN: 0360-0300. <https://doi.org/10.1145/356551.356554> (1969).
11. OpenCV. *OpenCV CPU optimizations* <https://github.com/opencv/opencv/wiki/CPU-optimizations-build-options>. (dostęp: 07.01.2024).
12. Keller, K. *i in. w Ullmann's Encyclopedia of Industrial Chemistry* (John Wiley & Sons, Ltd, 2000). ISBN: 9783527306732.

13. Dungarwal, S. *Understanding black & white film photography* <https://www.shreyansdungarwal.com/blog/2019/4/15/understanding-black-amp-white-film-photography>.
14. Dalladay, A. J. *The British Journal Photographic Almanac* 149–155 (Henry Greenwood & Co Ltd, London, 1956).
15. Blaker, A. *Photography: Art and Technique* ISBN: 9780716711162. <https://books.google.pl/books?id=bWdgQgAACAAJ> (W. H. Freeman, 1980).
16. BBC. *John Logie Baird - Historic figure* [https://www.bbc.co.uk/history/historic\\_figures/baird\\_logie.shtml](https://www.bbc.co.uk/history/historic_figures/baird_logie.shtml).
17. Analog. *Video Basics* <https://www.analog.com/en/technical-articles/basics-of-analog-video.html>.
18. Wigginton, C., Curran, M. & Brodeur, C. *Global mobile consumer trends* Deloitte. <https://www2.deloitte.com/content/dam/Deloitte/us/Documents/technology-media-telecommunications/us-global-mobile-consumer-survey-second-edition.pdf>.
19. McHugh-Johnson, M. *How Pixel's Super Res Zoom works* Google. <https://blog.google/products/pixel/super-res-zoom-google-pixel/>.
20. Cervantes, E. *What is Night Mode and how does it work* Android Authority. <https://www.androidauthority.com/what-is-night-mode-and-how-does-it-work-979590/>.
21. *Zrzut ekranu Adobe Photoshop* VisArts Center. <https://www.visartscenter.org/event/introduction-to-adobe-photoshop-2/>.
22. *Zrzut ekranu ImageJ* NC State University Libraries. <https://www.lib.ncsu.edu/workshops/introduction-to-imagej-for-scientific-research>.
23. *Batch Processing ImageJ*. <https://imagej.net/scripting/batch>.
24. Commons, W. *File:20221126\_17\_58\_02-Fiji (software) - Wikipedia.png* — Wikimedia Commons, the free media repository 2023. [https://commons.wikimedia.org/w/index.php?title=File:20221126\\_17\\_58\\_02-Fiji\\_\(software\)\\_-\\_Wikipedia.png&oldid=763353557](https://commons.wikimedia.org/w/index.php?title=File:20221126_17_58_02-Fiji_(software)_-_Wikipedia.png&oldid=763353557).
25. *Zrzut ekranu GIMP* GIMP. <https://www.gimp.org/release-notes/gimp-2.10.html>.

26. *Substance 3D Designer* Adobe. <https://helpx.adobe.com/substance-3d-designer.html>.

## **Spis rysunków**

1	Wywoływanie biało czarnej kliszy [13]. . . . .	5
2	Adobe Photoshop [21]. . . . .	7
3	ImageJ [22]. . . . .	8
4	Fiji - sukcesor ImageJ [24]. . . . .	9
5	GIMP wersja 2.10 [25]. . . . .	10
6	Adobe Substance Designer [26]. . . . .	11

## **Spis tabel**