

wspolne_dla_wszystkich/logo_uczelni.png

Poprawa jakości skanów zdjęć wykonanych techniką analogową

Raport II

projekt realizowany pod opieką prof. dr hab. inż. Artura Przelaskowskiego

wspolne_dla_wszystkich/logo_projektu.png

Spis treści

I. Raport 2

1. Wstęp	4
1.1. Wybór problemu	4
1.2. Cel projektu	4
1.3. Wykorzystywane materiały	4
1.4. Dobór zdjęć	5
2. Zdjęcia!	6
3. Wstępna analiza	18
3.1. Intensywność	19
3.1.1. Kod	19
3.1.2. Wyniki	20
3.2. Barwa, odcień	21
3.2.1. Kod	21
3.2.2. Wyniki	22
3.3. Saturacja	23
3.3.1. Kod	23
3.3.2. Wyniki	24
3.4. Kontrast	25
3.4.1. Kod	25
3.4.2. Wyniki	25
3.5. Wnioski i plany na przyszłość	26
4. Wykorzystywane narzędzia	26
5. Podział obowiązków	26

II. Raport 2

6. Cel projektu	28
7. Zdjęcia, zdjęcia!	28
8. Problemy	28
8.1. Niedoświetlenie	28
8.2. Zanieczyszczenia	31
9. Program i jego działanie	33
9.1. Tworzenie maski	33
9.1.1. Rozmycie gaussowskie	33
9.1.2. Wykonanie różnicy	34
9.1.3. Usunięcie ciemnych pikseli	34
9.1.4. Korekcja gamma	35
9.2. Gotowa maska	36
9.3. Działanie właściwe	37
10. Uwagi co do działania programu	38
11. Dostępność programu	38
12. Wykorzystywane narzędzia	39
13. Podział obowiązków	39

Część I

Raport 2

Streszczenie

Raport 1 projektu poprawy jakości zdjęć wykonanych analogowych przez grupę wtorkową z godziny 18 w składzie: Bartosz Wójcik, Katarzyna Szwed, Natalia Szymańska, Patrycja Szałajko, Aleksandra Wójcik, Karol Sęk, Michał Juskiewicz, Filip Sajko.

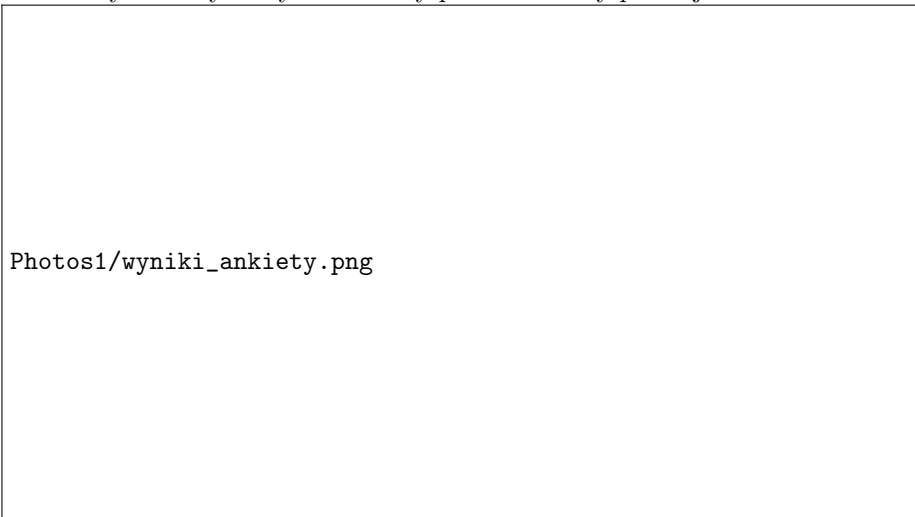
W tym raporcie skupimy się na opisie procesu wyboru tematu projektu, jego definiowaniu i planach jego analizy i rozwiązania. Ponadto wykonamy zdjęcia i zaczniemy je analizować.

1. Wstęp

1.1. Wybór problemu

Podczas wyboru tematu projektu kierowaliśmy się użytecznością i ponadczasowością naszej pracy oraz tym, żeby nasz program rozwiązywał problem, który zauważamy na co dzień. Zebraliśmy kilka naszych najlepszych pomysłów.

Były to: silnik szachowy, inteligentne rekomendacje muzyki, wydobywanie detali ze zdjęć o ekstremalnej jasności, opracowywanie wykresów związków chemicznych. Dodatkowo przeprowadziliśmy ankietę wśród wybranych 27 studentów naszej uczelni. Zapytaliśmy ich, rozwiązanie którego z powyższych problemów chcieliby zobaczyć. Wyniki ankiety przedstawiamy poniżej:



Photos1/wyniki_ankiety.png

Jak widać zdecydowanie największą popularnością cieszył się pomysł stworzenia programu poprawiającego jakość zdjęć.

1.2. Cel projektu

W ramach tego projektu skoncentrujemy się na opracowaniu metod umożliwiających stworzenie algorytmu, który normalizuje jasność zdjęć charakteryzujących się niedoświetleniem lub prześwietleniem. Na podstawie tego algorytmu planujemy opracować przyjazną użytkownikowi aplikację, dedykowaną poprawie jakości amatorskich fotografii. Odbiorcami naszego projektu są zarówno miłośnicy fotografii analogowej, jak i entuzjaści, którzy nie dysponują zaawansowanym sprzętem fotograficznym, a także wszyscy ci, którzy chcą wydobyć z archiwów rodzinnych nową jakość.

Fotografia analogowa, w przeciwieństwie do fotografii cyfrowej, nie pozwala na weryfikację efektów pracy tuż po wykonaniu zdjęcia. O błędach technicznych fotograf dowiaduje się dopiero po wywołaniu kliszy, co nie pozwala na wprowadzanie poprawek na bieżąco. Kluczowym aspektem wykonania poprawnej fotografii jest odpowiednie naświetlenie zdjęcia – aby uniknąć utraty detali w światłach i cieniach.

Naszym celem jest poprawianie jakości i wyciąganie szczegółów z zdjęć, w których te szczegóły zostały ukryte i utracone w wyniku niedoświetlenia i prześwietlenia.

1.3. Wykorzystywane materiały

Badania będziemy przeprowadzać z wykorzystaniem zdjęć zrobionych zarówno przy użyciu aparatu analogowego, jak i cyfrowego. Problemy, które dotyczą fotografii analogowej łatwo powtórzyć na aparacie cyfrowym ustawiając korektę ekspozycji. Celowo zrobimy zdjęcia jednego ujęcia o różnym poziomie naświetlenia tak, abyśmy mogli na nich testować algorytm wiedząc, jakich wyników powinniśmy się spodziewać. Dodatkowo zależy nam na pozyskaniu informacji w

jaki sposób zdjęcia doświetlone niepoprawnie odstają od poprawnej ekspozycji. Zależy nam, aby zdjęcia były o różnorodnej tematyce – od zdjęć natury przez portrety po zdjęcia architektury tak, aby mieć pewność, że nasza metoda ma szerokie zastosowanie w życiu codziennym.

1.4. Dobór zdjęć

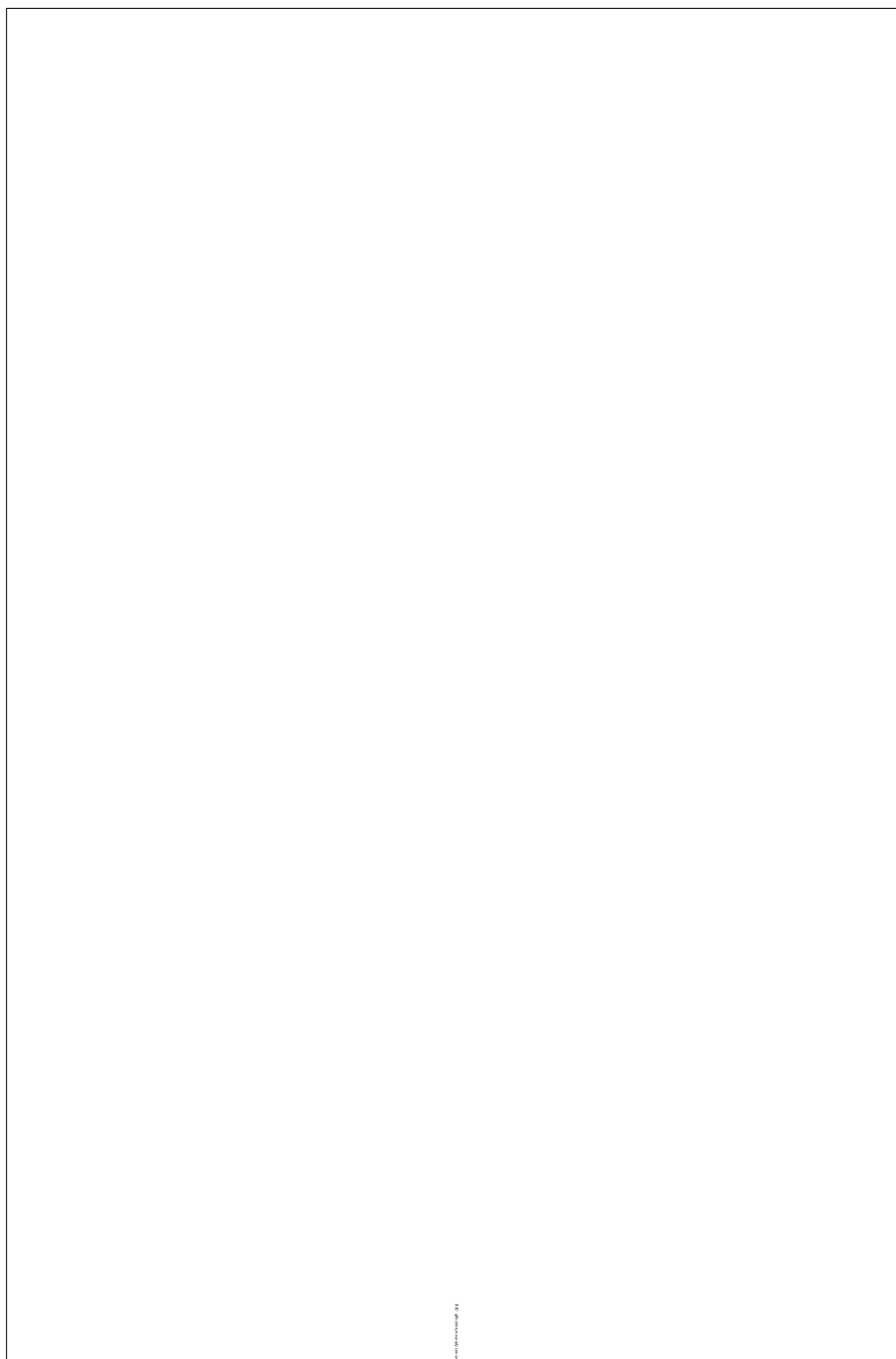
Będziemy pracować na wysokiej jakości cyfrowych skanach filmu zdjęć analogowych, a także na typowych dla nas cyfrowych zdjęciach. Posłużymy się archiwalnymi zdjęciami znalezionymi w rodzinnych albumach, naszych własnych kolekcjach i wykonanymi celowo na potrzeby tego projektu.

W tym celu kilkoro członków naszego zespołu chwyciło za aparaty, i ruszyło fotografować świat!

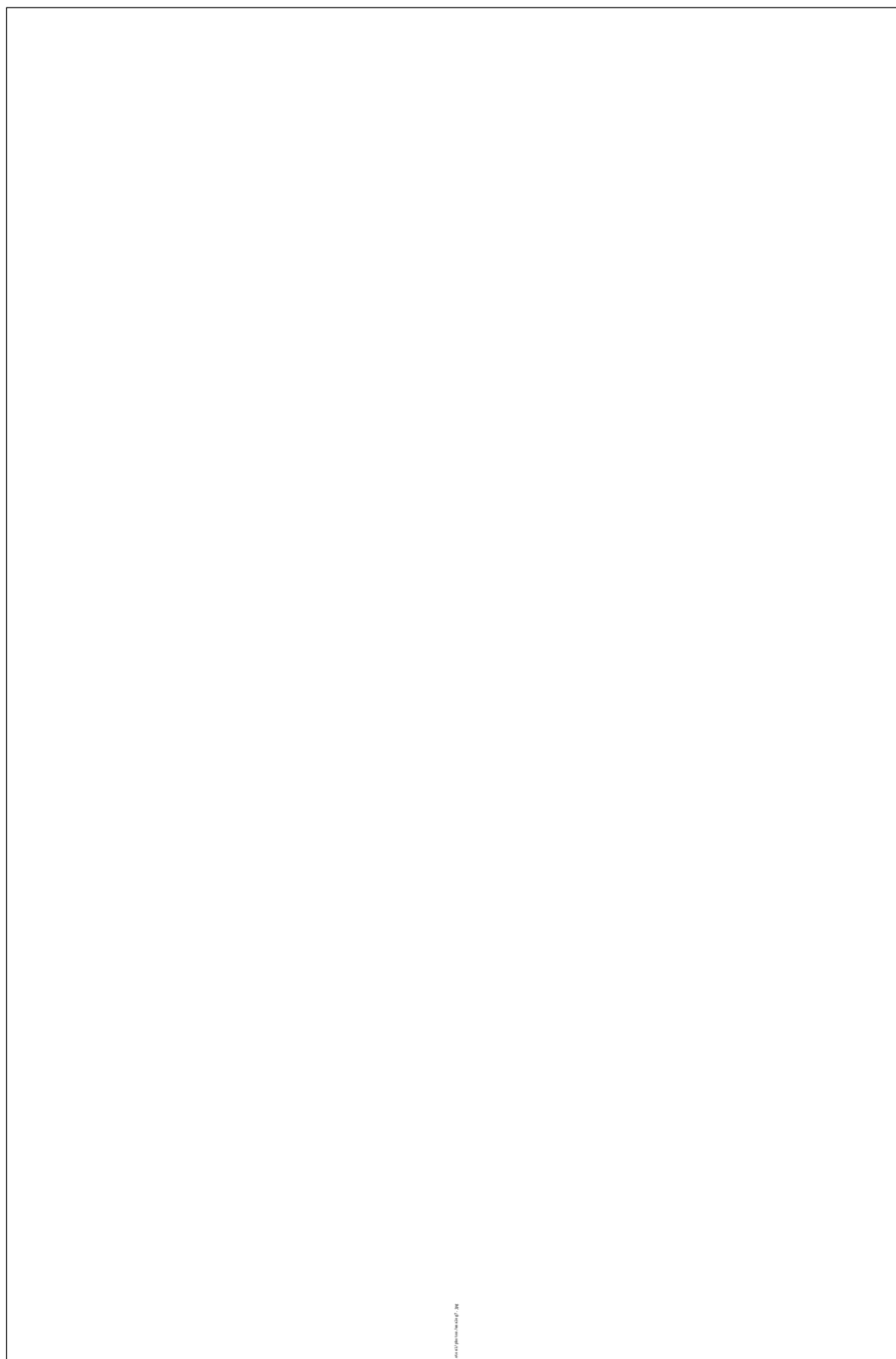
2. Zdjęcia!

Kilka przykładowych zdjęć spośród zebranych przez nas, które obrazują problem utraty szczegółów zdjęć analogowych w przypadku ich niedoświetlenia.

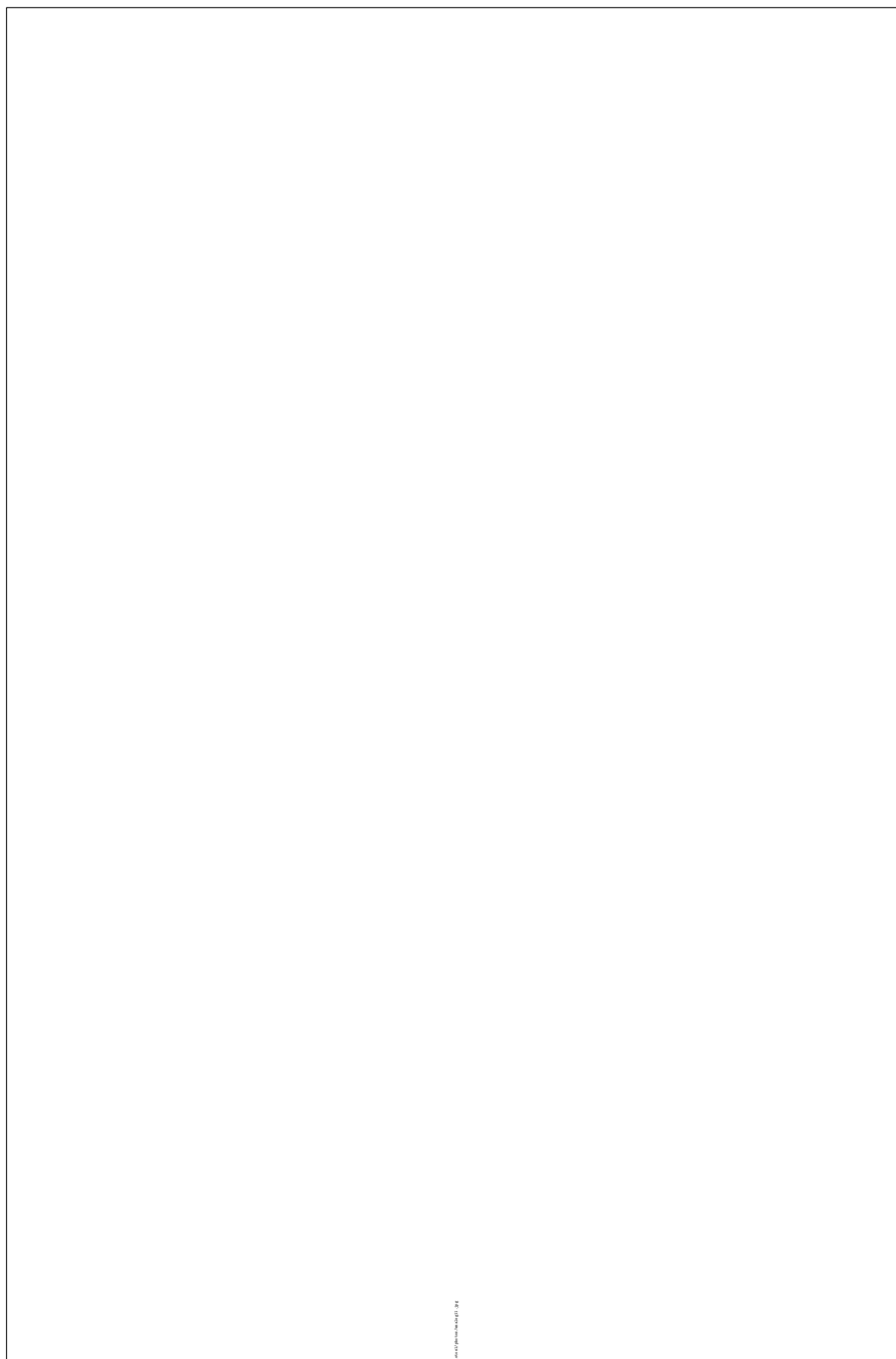
Dalej znajdują się także porównania kilku wybranych zdjęć wykonanych aparatem cyfrowym przy różnych ustawieniach ekspozycji. Dla zdjęć cyfrowych najwięcej detali traconych jest w prześwietlonych punktach.



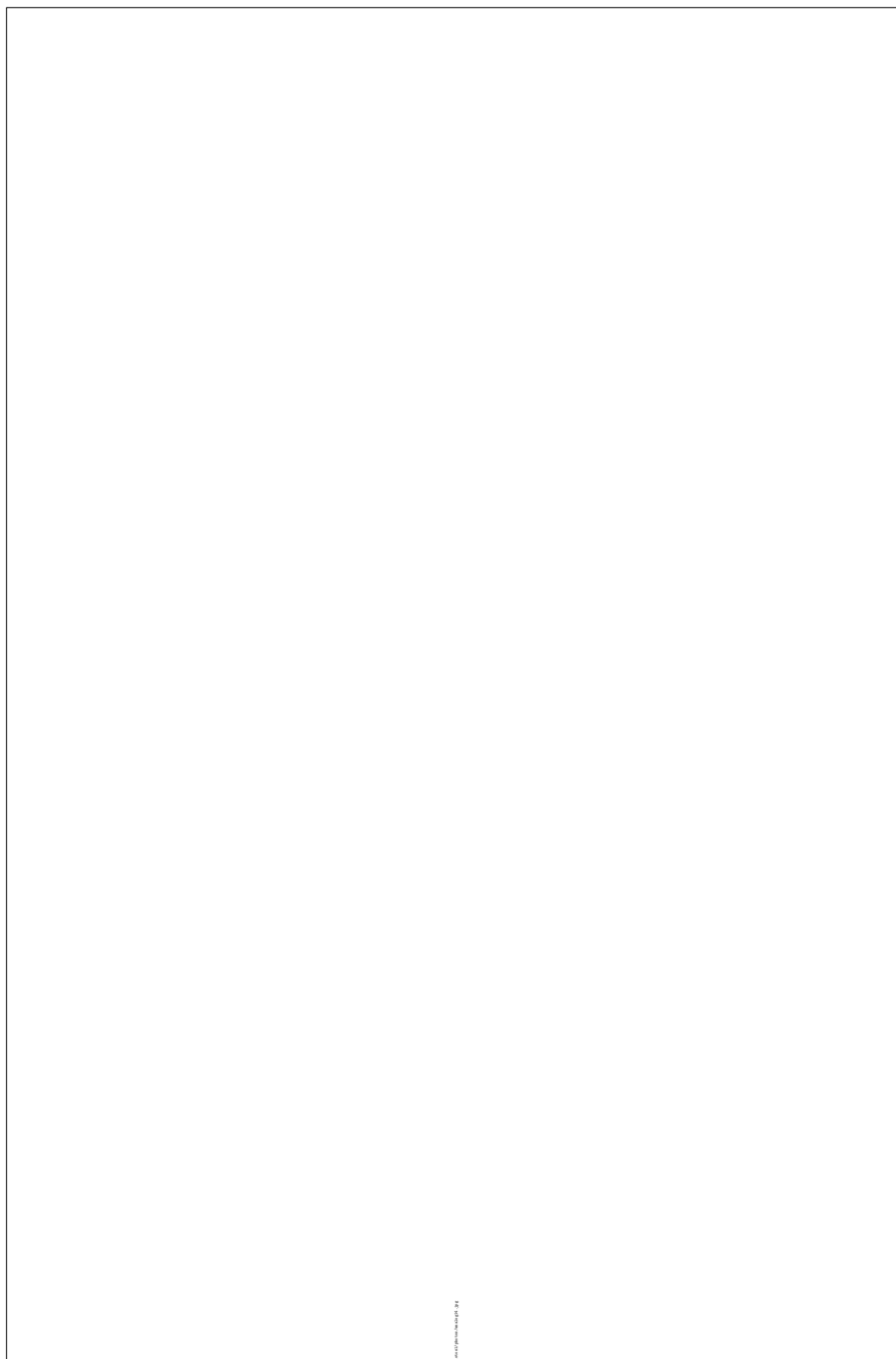
Rysunek 1: Zdjęcie niedoświetlone



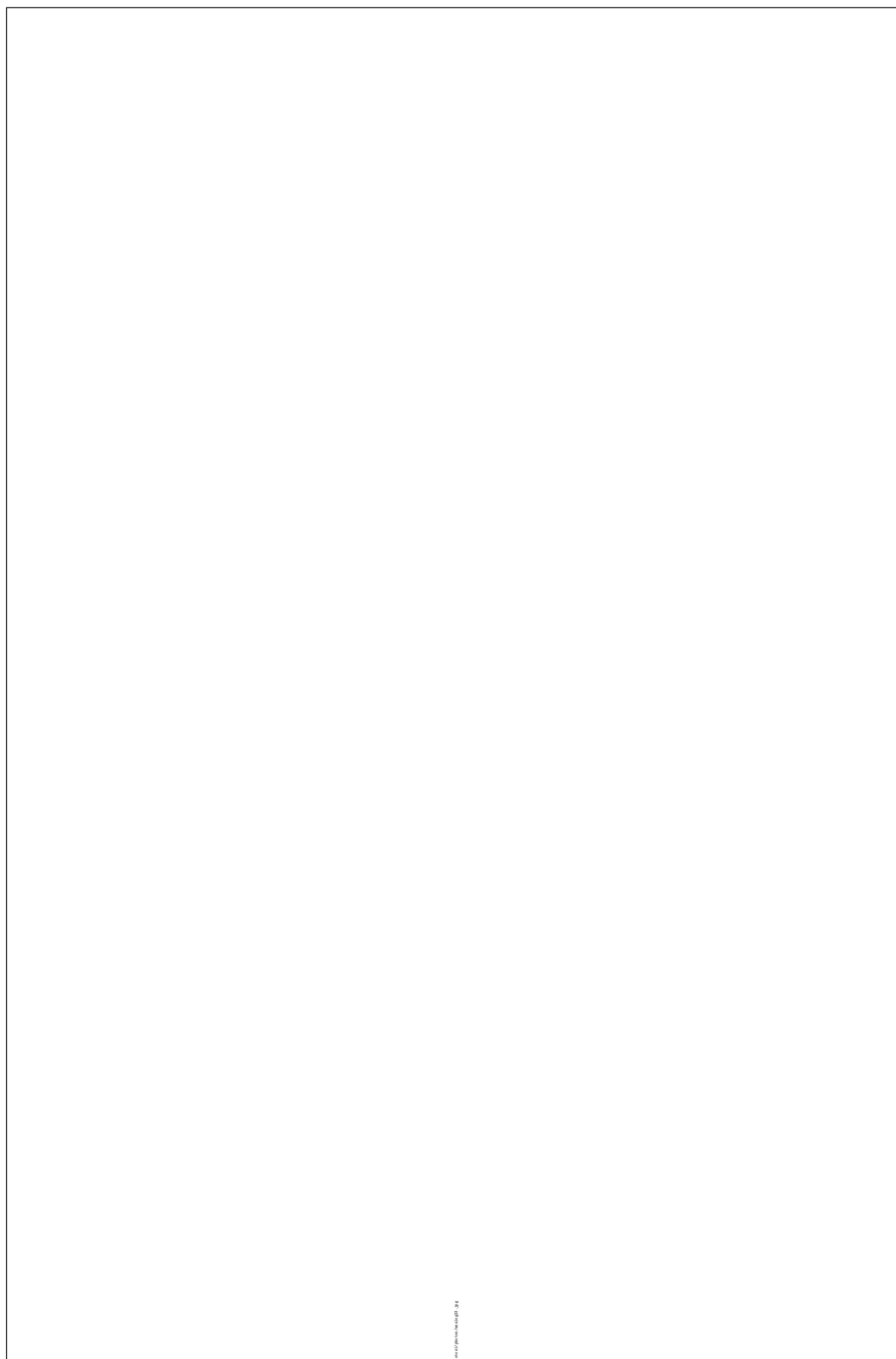
Rysunek 2: Zdjęcie doświetlone



Rysunek 3: Zdjęcie niedoświetlone



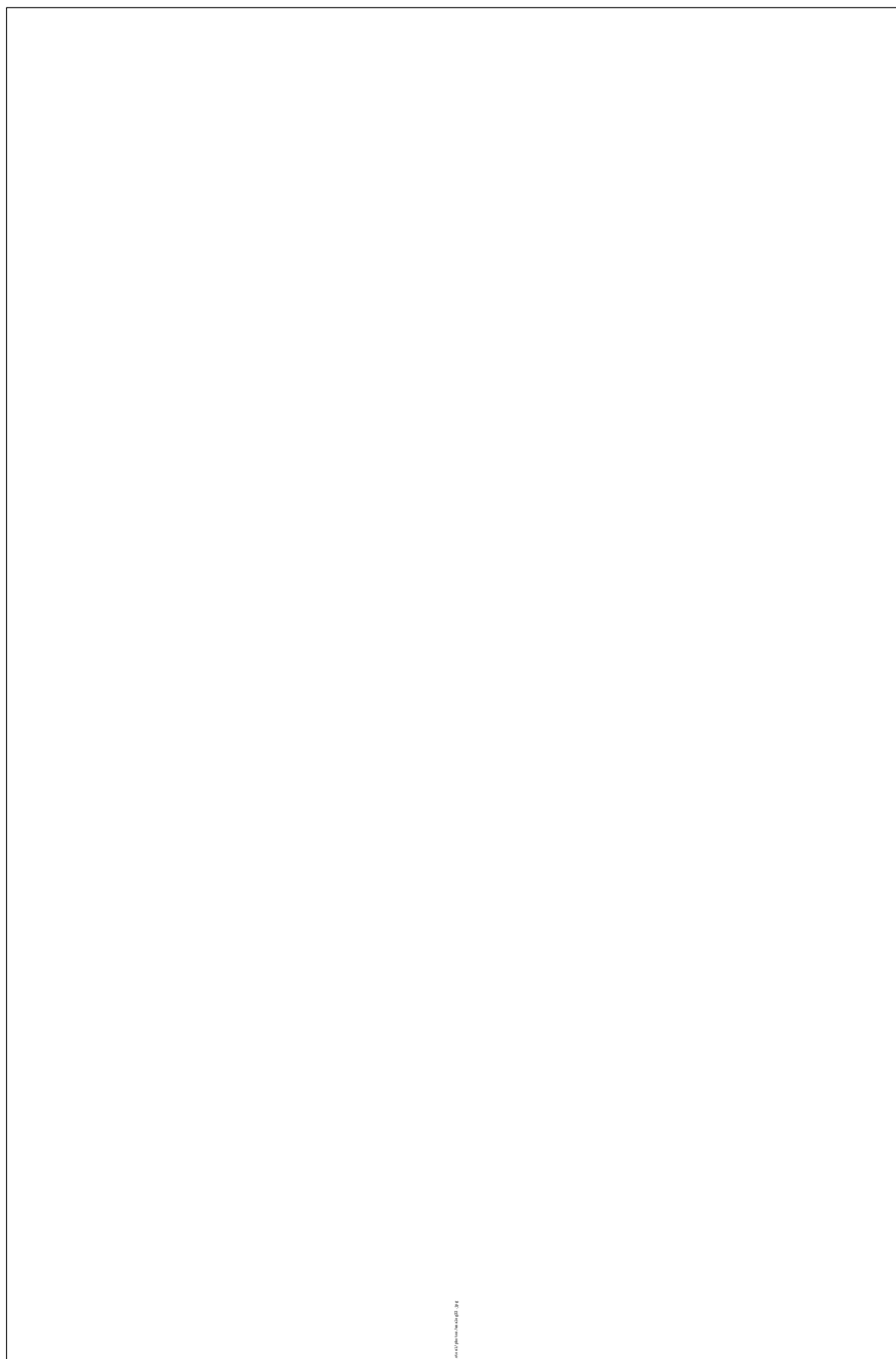
Rysunek 4: Zdjęcie doświetlone



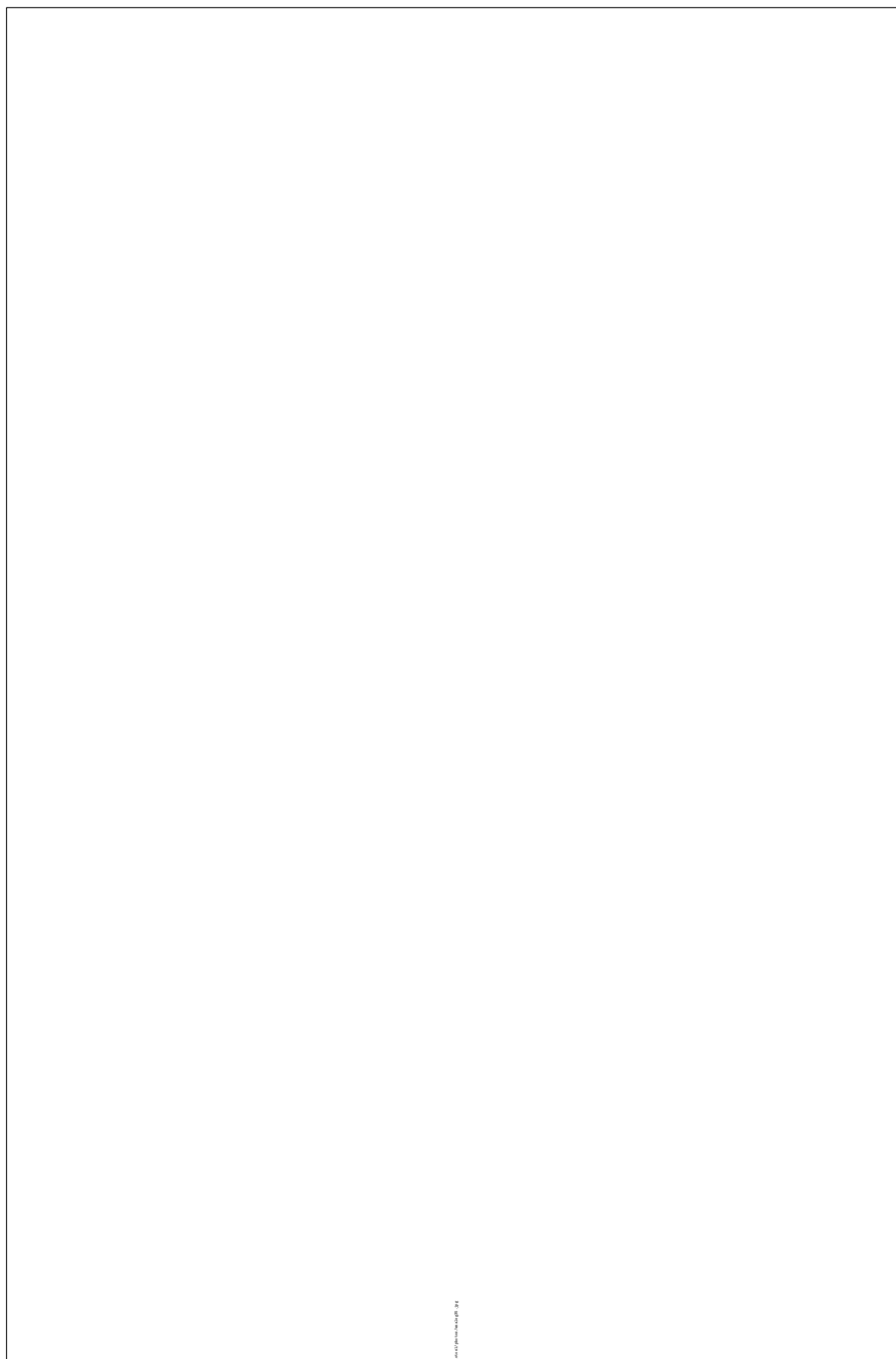
Rysunek 5: Zdjęcie niedoświetlone

wspolne_dla_wszystkich/logo_projektu.png
Poprawa jakości zdjęć

wspolne_dla_wszystkich/logo_uczelni.png



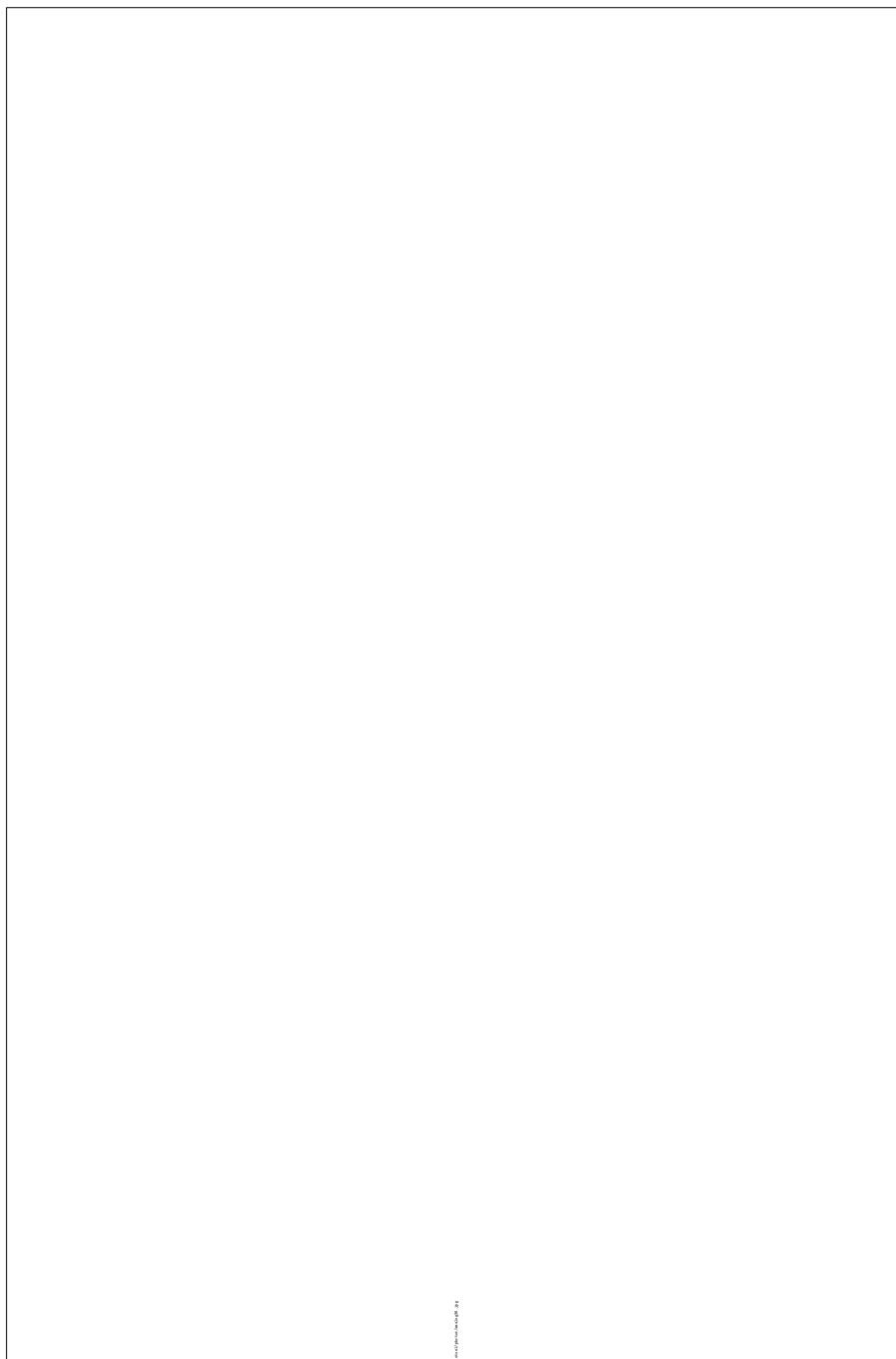
Rysunek 6: Zdjęcie doświetlone



Rysunek 7: Zdjęcie niedoświetlone

wspolne_dla_wszystkich/logo_projektu.png
Poprawa jakości zdjęć

wspolne_dla_wszystkich/logo_uczelni.png



Rysunek 8: Zdjęcie doświetlone

A także w przypadku fotografii cyfrowej:



Rysunek 9: Zdjęcie niedoświetlone



Rysunek 10: Zdjęcie doświetlone

wspolne_dla_wszystkich/logo_projektu.png
Poprawa jakości zdjęć

wspolne_dla_wszystkich/logo_uczelni.png

Photos1/photos/balony1.jpg

Rysunek 11: Zdjęcie prześwietlone

Photos1/photos/kot3.jpg

Rysunek 12: Zdjęcie niedoświetlone



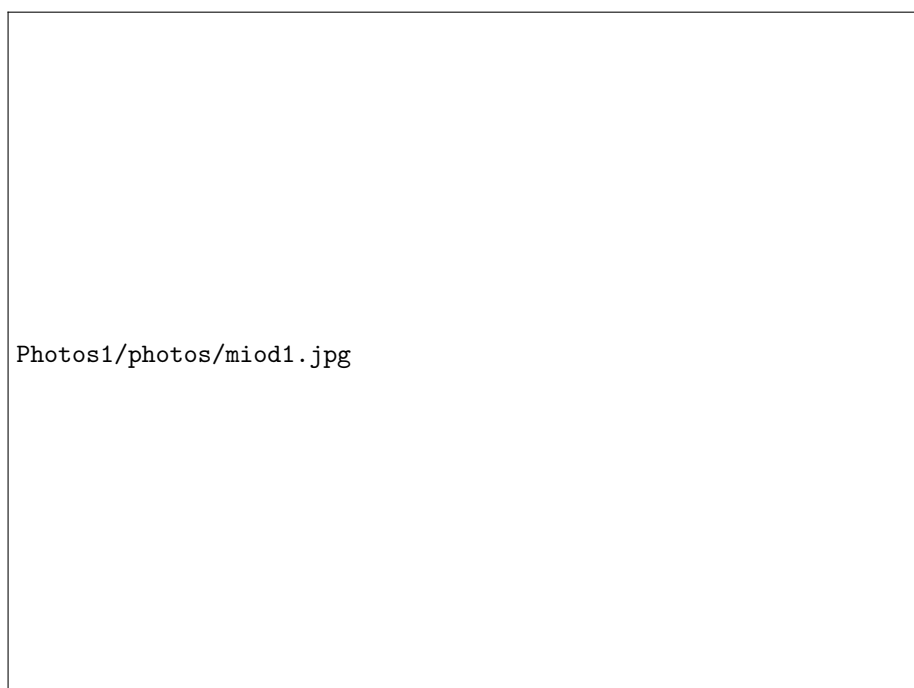
Rysunek 13: Zdjęcie doświetlone




Rysunek 14: Zdjęcie prześwietlone



Rysunek 15: Zdjęcie niedoświetlone



Rysunek 16: Zdjęcie doświetlone



Photos1/photos/miod2.jpg

Rysunek 17: Zdjęcie prześwietlone

Spora (i ciągle rosnąca) baza zdjęć analogowych i cyfrowych dogłębnie ilustrujących sedno problemu z którym się zmagamy znajduje się tutaj: https://drive.google.com/drive/folders/1SFu_A46nBXhL19diXnxV-iyEyH0fnhFA

3. Wstępna analiza

Jakkolwiek często ‘na oko’ względnie łatwo porównując dwa zdjęcia wskazać to, któremu brakuje szczegółów lub zostało niedoświetlone, to warto się zastanowić co tak właściwie znaczy to, że zdjęcie jest niedoświetlone.

W celu analizy i próby zdefiniowania zdjęcia o skrajnej jasności utworzyliśmy kilka Matlabowych skryptów, które analizują wybrane parametry zdjęć.

(Uwaga techniczna) W samym raporcie posługujemy się przykładem kilku ujęć jednego zdjęcia aby zademonstrować nasze działania, a także aby zachować czytelność. Całość dostępna jest tutaj: https://drive.google.com/drive/folders/1SFu_A46nBXhL19diXnxV-iyEyH0fnhFA.

3.1. Intensywność

W przypadku analizy zdjęcia intensywność odnosi się do jasności piksela i opisuje ogólną ilość światła w obrazie. Jest to miara luminancji, czyli składowej jasności obrazu, niezależna od koloru. Utworzyliśmy histogramy pokazujące rozłożenie jasności pikseli na obrazach. Na podstawie tych wykresów można wysunąć wnioski na temat tego czy zdjęcie jest dobrze oświetlone, niedoświetlone czy prześwietlone. Zauważyliśmy, że zebrane przez nas zdjęcia analogowe są znacznie ciemniejsze od tych cyfrowych. Histogramy zdjęć robionych w tych samych warunkach, ale z inną ekspozycją znacząco różnią się między sobą, jednak ta różnica jest najbardziej widoczna w przypadku zdjęć cyfrowych.

3.1.1. Kod

```
files = dir("photos\*.jpg");

for i = 1:length(files)

    clear count g G im k light max n s x y;
    im = imread(strcat("photos\ ", files(i).name));
    g = rgb2gray(im);
    G = g(:);
    s = length(G);
    figure(1);
    set(gcf, 'Units', 'Normalized', 'OuterPosition', [0 0 1 1]); %wielkość okna

    subplot(1,3,[1,2]);
    histogram(G,'FaceColor', 'ffffff');

    [count, n] = histcounts( G, 255 );
    max = max(count);
    max = max*1.2;

    xlim([0 255]);
    ylim([0,max]);

    grid on;
    title('Histogram jasności pikseli','FontSize', 30);
    xlabel('Wartość jasności','FontSize',20);
    ylabel('Ilość pikseli','FontSize',20);

    x = [0 0 0 0];
    y = [0 0 max max];
    light = 0.66;
    for k =1:1:5
        hold on;
        x = x + [0 51 51 0];
        patch(x,y,'k','FaceAlpha',light);
        x = x + [51 0 0 51];
        light = light *0.66;
    end

    subplot(1,3,3);
    imshow(im);

    exportgraphics(gcf, strcat("intensity\ ",
        \\files(i).name(1:length(files(i).name)-4), "_intensity.jpg"))

    close;
end
```

3.1.2. Wyniki

Photos1/intensity/jagier1_intensity.jpg

Photos1/intensity/jagier2_intensity.jpg

Photos1/intensity/jagier3_intensity.jpg

3.2. Barwa, odcień

Hue – z angielskiego coś pomiędzy barwą a odcieniem w fotografii odnosi się do podstawowego koloru światła, czyli pozycji danego koloru w spektrum barw widzialnych. Wyrażany jest w stopniach od 0 do 360° na kole barw. Natomiast zmiana hue tylko przesuwa kolor, ale nie zmienia jego jasności ani nasycenia. Nie badaliśmy wartości hue dla zebranych przez nas zdjęć analogowych, ponieważ są czarno-białe. Analiza hue dla zdjęć cyfrowych pokazała nieznaczne różnice przy różnej ekspozycji.

3.2.1. Kod

```
files = dir("photos\*.jpg");

for i = 1:length(files)

    clear count g G im max n;
    im = imread(strcat("photos\ ", files(i).name));
    g = rgb2hsv(im);
    g = g(:, :, 1);
    g = g*255;
    G = g(:);
    figure(1);
    set(gcf, 'Units', 'Normalized', 'OuterPosition', [0 0 1 1]); %wielkość okna
    subplot(1,3,[1,2]);
    histogram(G,'FaceColor', 'ffffff');

    [count, n] = histcounts( G, 255 );
    max = max(count);
    max = max*1.2;

    xlim([0 255]);
    ylim([0,max]);

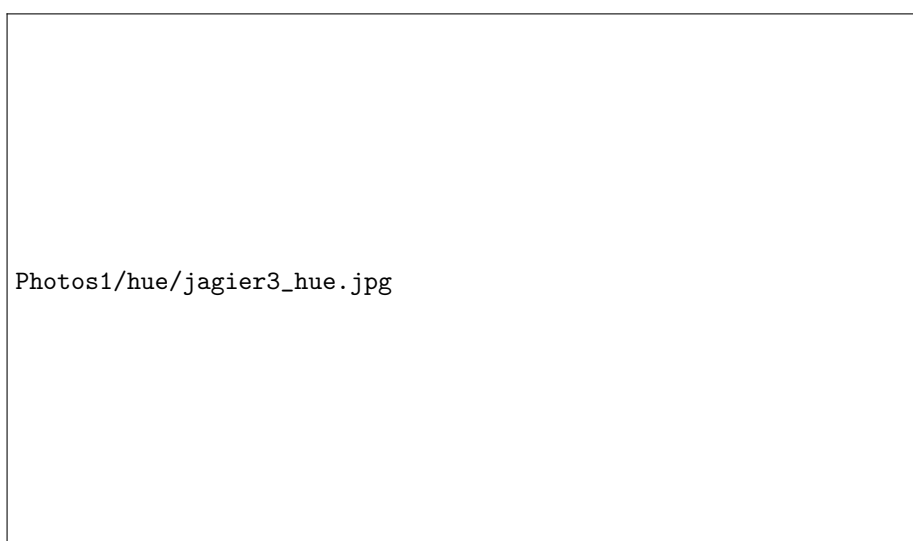
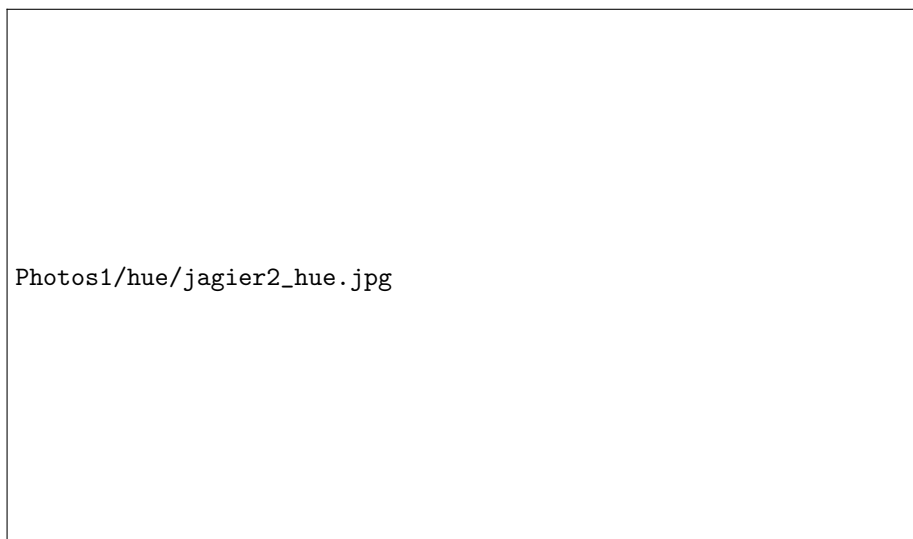
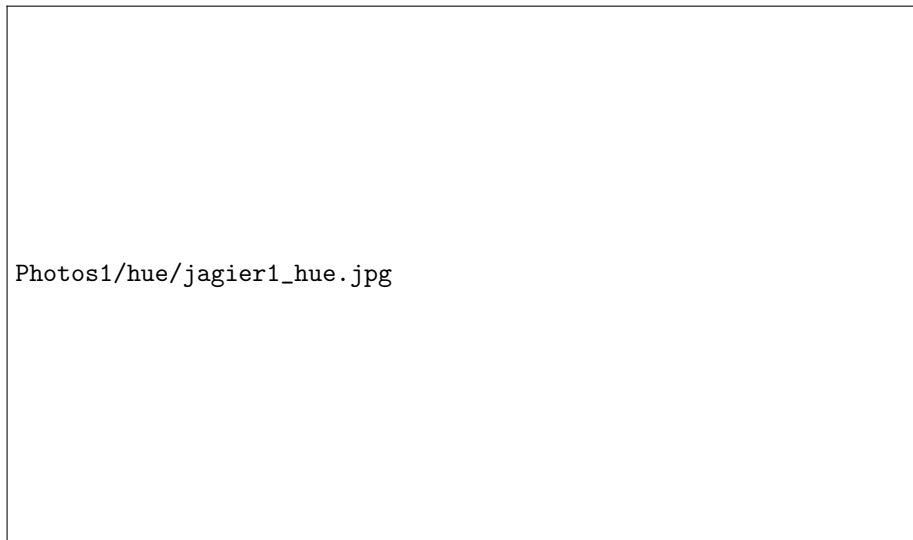
    grid on;
    title('Histogram odcienia pikseli','FontSize', 30);
    xlabel('Wartość odcienia','FontSize',20);
    ylabel('Ilość pikseli','FontSize',20);

    subplot(1,3,3);
    imshow(im);

    exportgraphics(gcf, strcat("hue\ ",
                                "\\files(i).name(1:length(files(i).name)-4), "_hue.jpg"))
    close;

end
```

3.2.2. Wyniki



3.3. Saturacja

Saturacja w fotografii to stopień intensywności kolorów na zdjęciu. Określa, jak bardzo barwy są nasycone – od wyblakłych i niemal czarno-białych (niska saturacja) do bardzo żywych i intensywnych (wysoka saturacja). Tak samo jak z hue nie badaliśmy saturacji dla zdjęć czarno-białych. Można zauważyć znaczne różnice w saturacji zdjęć cyfrowych w zależności od stopnia naświetlenia.

3.3.1. Kod

```
files = dir("photos\*.jpg");

for i = 1:length(files)

    clear count g G im max n;
    im = imread(strcat("photos\", files(i).name));
    g = rgb2hsv(im);
    g = g(:,:,2);
    g = g*255;
    G = g(:);
    figure(1);
    set(gcf, 'Units', 'Normalized', 'OuterPosition', [0 0 1 1]); %wielkość okna
    subplot(1,3,[1,2]);
    histogram(G,'FaceColor', '#ffffff');

    [count, n] = histcounts( G, 255 );
    max = max(count);
    max = max*1.2;

    xlim([0 255]);
    ylim([0,max]);

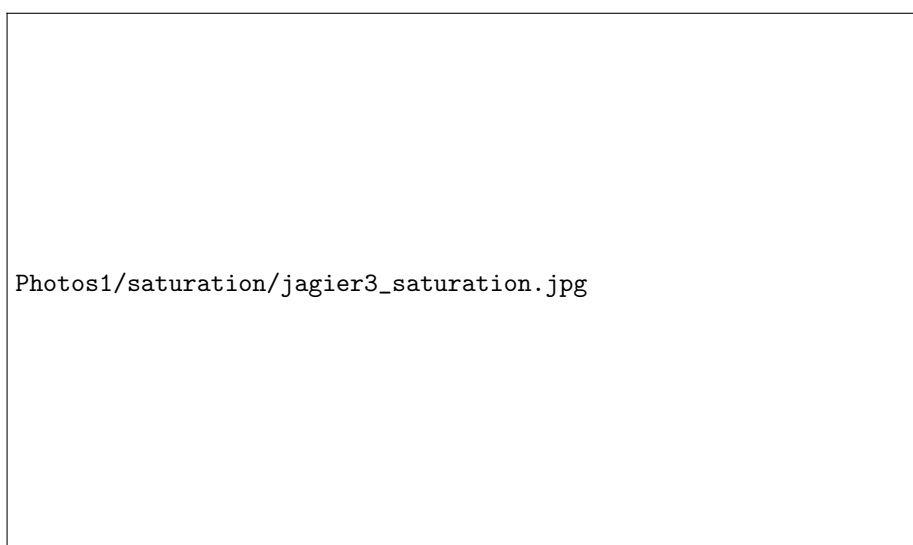
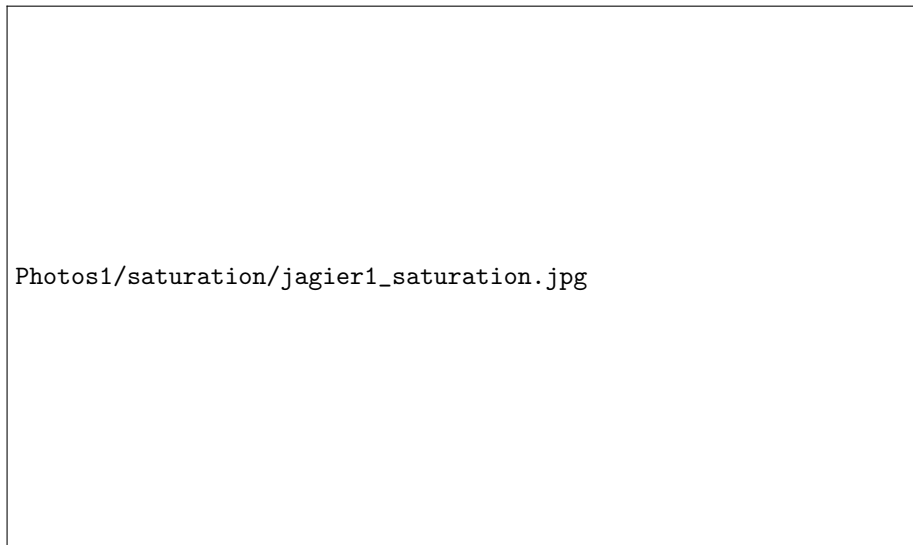
    grid on;
    title('Histogram nasycenia pikseli','FontSize', 30);
    xlabel('Wartość nasycenia','FontSize',20);
    ylabel('Ilość pikseli','FontSize',20);

    subplot(1,3,3);
    imshow(im);

    exportgraphics(gcf, strcat("saturation\","
                               "\\files(i).name(1:length(files(i).name)-4),"_saturation.jpg"))
    close;

end
```

3.3.2. Wyniki



3.4. Kontrast

Kontrast w fotografii to różnica między jasnymi i ciemnymi obszarami zdjęcia. Określa, jak bardzo elementy obrazu różnią się od siebie pod względem jasności, koloru lub tonu. Wyższy kontrast sprawia, że zdjęcie wygląda bardziej dynamicznie, a niski kontrast daje bardziej miękkie, wyblakły efekt.

3.4.1. Kod

```
files = dir("photos\*.jpg");
cont = struct('plik', cell(1,length(files)), 'kontrast',
             '\cell(1,length(files)), 'jasnosc', cell(1,length(files)));

for i = 1:length(files)

clear g G l im j k m n R;
im = imread(strcat("photos\", files(i).name));
g = rgb2gray(im);
g = single(g);
g=g/255;
G = g(:);
I = mean(G);
[n, m] = size(g);
R =0.0;

for k=1:1:n
    for j = 1:1:m
        R = R + (g(k,j) - I)^2;
    end
end

R = R/(m*n);

R = sqrt(R);

%disp(["Kontrast zdjęcia" R]);
%disp(["Średnia jasność zdjęcia" I]);

cont(i).plik = files(i).name;
cont(i).kontrast = R;
cont(i).jasnosc = I;

end

writetable(struct2table(cont), 'contrast.csv')
```

3.4.2. Wyniki

Co w przypadku naszego zdjęcia dało wyniki następujące:

Zdjęcie	Kontrast	Jasność
Zdjęcie stołu niedoświetlone	0.1856783	0.2637069
Zdjęcie stołu idealne	0.2608515	0.39802570
Zdjęcie stołu prześwietlone	0.3019148	0.54119000

Tabela 1: Wyniki dla używanego w raporcie przykładowego zdjęcia stołu

3.5. Wnioski i plany na przyszłość

Analiza danych zebranych w powyższych częściach okazała się trudnym zadaniem, dlatego wrócimy do niej w kolejnym raporcie. Zamieszczamy jednak poniższą tabelę, której poprawności statystycznej dla ogółu danych nie jesteśmy pewni. Na podstawie danych jesteśmy wciąż w stanie postawić hipotezę, że: im większe doświetlenie tym wyższy kontrast i jasność. nirrsze

EV	-1	+ δ	0	+ δ	+1
Kontrast	0.119	+37 %	0.163	+26 %	0.206
Jasność	0.231	+27 %	0.294	+20 %	0.352

Tabela 2: Średnie jasności i kontrastu od korekty naświetlenia

4. Wykorzystywane narzędzia

W tej części naszego projektu korzystaliśmy z następujących narzędzi:

- Programu Matlab – do analizy zdjęć;
- Programu LibreOffice Calc – do analizy wyników ankiety;
- \LaTeX 2 ϵ – do przygotowania raportu;
- Google Drive – do udostępniania plików;
- 7zip – do kompresji zdjęć;
- Aparatów:
 - Canon EOS 300 z obiektywem Tamron 28-105mm 1:4-5.6 i kliszą Foma-pan 400
 - Fujifilm FinePix L55 Digital Camera – Black (12MP, 3x Optical Zoom)

5. Podział obowiązków

Po wyborze celu projektu wszyscy zajęliśmy się zdobywaniem wiedzy na temat problemów fotografii analogowej, a także możliwych poprawy jakości zdjęć.

Posiadając wstępną wiedzę na temat materii projektu organicznie wstępnie podzieliliśmy się zajęciami zgodnie z naszymi zainteresowaniami:

- pozyskiwanie materiałów testowych – Aleksandra Wójcik, Bartosz Wójcik;
- opracowanie skryptów do analizy zdjęć i zbieranie informacji do algorytmu – Katarzyna Szwed, Karol Sęk, Michał Juskiewicz;
- opracowywanie raportu – Patrycja Szałajko, Natalia Szymańska, Filip Sajko.

Część II

Raport 2

Streszczenie

Raport 2 projektu poprawy jakości cyfrowych skanów zdjęć wykonanych techniką analogową przez grupę nr 9 (wtorkową z godziny 18) w składzie: Bartosz Wójcik, Katarzyna Szwed, Natalia Szymańska, Patrycja Szałajko, Aleksandra Wójcik, Karol Sęk, Michał Juskiewicz, Filip Sajko.

W tym raporcie zdefiniujemy cel naszego projektu i opiszemy problem z którym się mierzymy. Przedstawimy ponadto wstępną wersję naszego programu i zademonstrujemy jego skuteczność.

6. Cel projektu

W związku ze słusznymi uwagami i wskazówkami, podjęliśmy decyzję o ukonkretyzowaniu celu naszego projektu. Skupimy się przede wszystkim na poprawianiu defektów cyfrowych skanów zdjęć analogowych. Staramy się trafić do dwóch (niekoniecznie rozłącznych) grup osób – współczesnych fanów fotografii analogowej (będącą dla amatora niełatwą sztuką) i posiadaczy pękaty archiwów zdjęć rodzinnych chcących je zachować i cyfrowo utrwalić.

7. Zdjęcia, zdjęcia!

Profilowym zdjęciem dla nas jest portret – tak indywidualny jak i grupowy. Jest to typ zdjęć najbardziej popularny w rodzinnych albumach – mnogość w nich zdjęć z ważnych dla danej familii wydarzeń: chrztów, wesel czy pogrzebów... Służą one utrwaleniu wspomnień oraz pamięci po krewnych i bliskich, którzy już odeszli... A więc noszących dużą wartość emocjonalną dla ich posiadacza.

Przykładem takiej osoby jest nasza koleżanka Ola – wraz z jej rodzinnym albumem.

8. Problemy

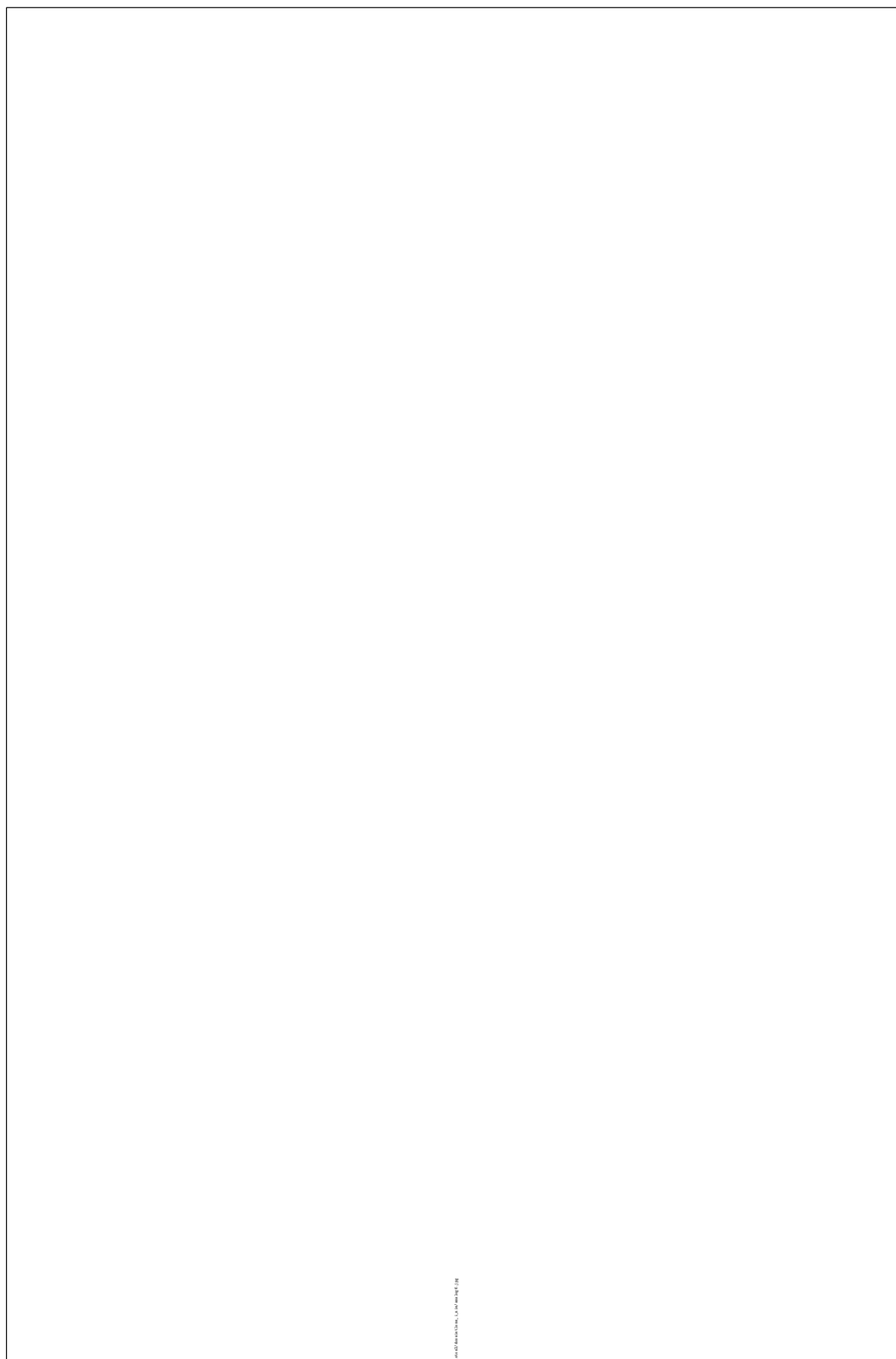
Wykonywanie, a następnie ‘ucyfrowienie’ zdjęcia w technice analogowej wiąże się z różnymi trudnościami, które mogą znacząco obniżyć jakość zdjęcia – a z tym satysfakcję jego posiadacza. Głównymi problemami, którym będziemy przeciwdziałać, będą niedoświetlenie zdjęcia i zanieczyszczenia powietrza osadzające się na oryginalnym zdjęciu i skanerze podczas procesu zmiany informacji z analogowej na cyfrową.

8.1. Niedoświetlenie

Niedoświetlenie jest problemem trudnym – zwłaszcza dla fanów-amatorów techniki analogowej. Zasadnicza większość klasycznych aparatów nie posiada zaawansowanej mechaniki automatycznie wybierającej odpowiednie ustawienia aparatu, a brak możliwości podglądu tego, jak dane zdjęcie wyszło, często doprowadza do sytuacji, gdzie po wielu dniach okazuje się, że na zdjęciu chwili, którą fotograf chciał uchwycić i utrwalić, niewiele widać, bo przez złe ustawienia większość szczegółów jest niewidoczna...¹

Dla przykładu przypomnijmy:

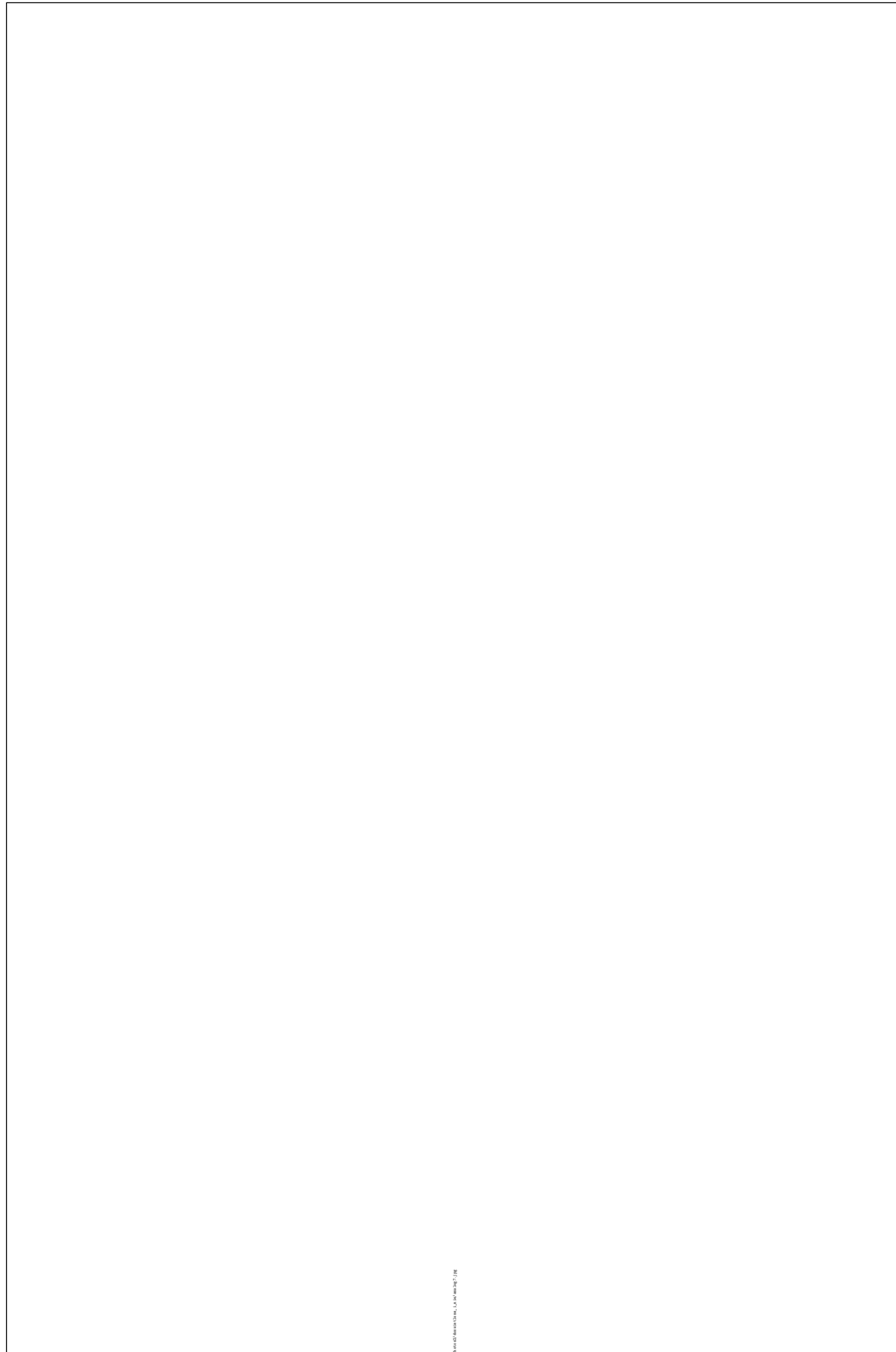
¹ Jest to problem, który szeroko wraz z przykładami i analizą numeryczną opisywaliśmy w raporcie pierwszym.



Rysunek 18: Zdjęcie niedoświetlone.

wspolne_dla_wszystkich/logo_projektu.png
Poprawa jakości zdjęć

wspolne_dla_wszystkich/logo_uczelni.png



Rysunek 19: ...i to w punkt.

8.2. Zanieczyszczenia

Prawie że cała nasza² codzienność dzieje się w niesterylnych warunkach. I jakkolwiek dla większości z nas nie jest to problem, są miejsca i sytuacje, gdy prowadzi to do pewnych problemów. W powietrzu nas otaczającym jest pełno unoszących się zanieczyszczeń: włosów, kurzu, futra etc.

Problematyczne jest natomiast osadzanie się wspomnianej powyżej materii na zdjęciach i soczewkach – która przenosi się na skan tworząc nieestetyczne artefakty:



Rysunek 20: W skrajnych wypadkach może wyglądać to nawet tak.



Rysunek 21: Choć bardziej częstym jest ten przypadek.

² nie jesteśmy wszak ani naukowcami, ani lekarzami.



Rysunek 22: A także taki.

9. Program i jego działanie

Zbrojni w wiedzę co chcemy osiągnąć i zapas zebranych skanów zdjęć do testów wzięliśmy się do pracy nad programem. Stworzyliśmy zaawansowany program, który w przypadku funkcjonalności ‘anty-zanieczyszczeniowej’ inteligentnie przeszukuje cały obszar zdjęcia, zaznacza artefakty, a w następnej fazie działania usuwa je.

Przechodząc do szczegółów, działanie programu można opisać za pomocą kilku kolejnych faz działania³ na przykładzie poniższego zdjęcia:



Rysunek 23: Przykładowe zdjęcie – to za jego pomocą opiszymy działanie programu.

9.1. Tworzenie maski

9.1.1. Rozmycie gaussowskie

Pierwszym krokiem generowania maski jest stworzenie rozmycia gaussowskiego zdjęcia.

Wygładzanie gaussowskie jest efektem rozmywania obrazu za pomocą funkcji Gaussa, która jest szeroko wykorzystywana w grafice komputerowej w celu uzyskania gładkiego wygładzenia obrazu i wyciszenia szumu informacyjnego.

Za pomocą funkcji danej wzorem dla każdego piksela:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (1)$$

gdzie x, y to współrzędne danego piksel'a, a σ oznacza odchylenie standardowe.⁴

³ Algorytm przekształca zdjęcie z formatu RGB do formatu HSV (Hue, Saturation, Value) i działa tylko na Value, która definiuje jasność piksela w skali 0-255.

⁴ Źródło: https://en.wikipedia.org/wiki/Gaussian_blur

Photos2/gauss_blurr/gauss_blurr_gpt1.png

Rysunek 24: Zdjęcie po wykonaniu rozmycia gaussowskiego.

9.1.2. Wykonanie różnicy

Następnie odejmujemy od oryginału zdjęcia otrzymane w poprzednim etapie rozmycia. Pozwala to na wykrycie najbardziej kontrastowych elementów zdjęcia.

Photos2/difference/difference_gpt1.png

Rysunek 25: Zdjęcie po wykonaniu różnicy.

9.1.3. Usunięcie ciemnych pikseli

Zostawiamy tylko jasne piksele (według standardowych ustawień jest to jasność powyżej 60) i nadajemy im maksymalną wartość 255. Pozostałym pikselom ustawiamy jasność na 0.

9.1.4. Korekcja gamma

Korekcja gamma jest techniką stosowaną w grafice komputerowej, której celem jest dostosowanie jasności obrazu do ludzkiego nielinowego postrzegania światła. Korekcja gamma kompensuje ten nieliniowy sposób widzenia, pozwalając na efektywniejsze wykorzystanie dostępnych poziomów jasności.

Korekcja gamma dokonuje transformacji jasności w następujący sposób:

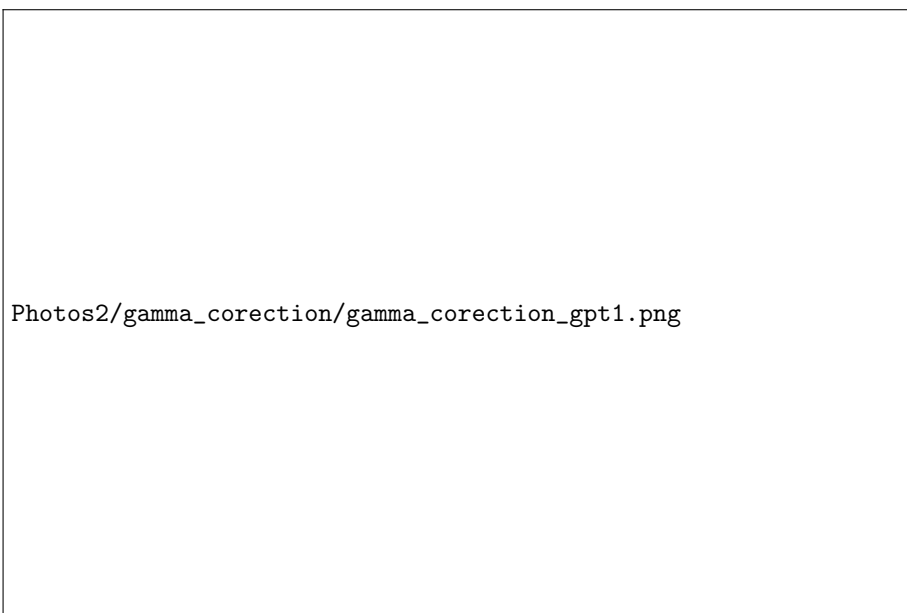
$$L'(x, y) = L(x, y)^\gamma \quad (2)$$

gdzie:

- γ to współczynnik gamma,
- $L(x, y)$ to wartości jasności pikseli.

Wiedząc to, w tym etapie bierzemy ponownie oryginalne zdjęcie, wykonujemy na nim korekcję gamma, a następnie na tym zdjęciu wykonujemy etapy 1-3.

Działanie to pozwala uwzględnić jak najwięcej artefaktów – znajdujących się także na jasnym tle.



Rysunek 26: Po wykonaniu korekcji gamma i etapów 1-3.

9.2. Gotowa maska

Po tym wszystkim otrzymujemy dwie podmaski (jedna robiona na oryginalnym zdjęciu, a druga na jaśniejszym – rozświetloną modulacją gamma). Końcową maskę otrzymujemy biorąc wszystkie znalezione piksele z obu podmasek.



Rysunek 27: Maska wykonana z oganianego zdjęcia



Rysunek 28: Maska wykonana z zdjęcia rozświetlonego korekcją gamma

9.3. Działanie właściwe

Mając wygenerowaną maskę właściwą, przechodzimy po wyznaczonych przez nią pikselach na oryginalnym zdjęciu. Dla każdego piksela w zaznaczonego w masce wyliczamy nową wartość jasności – średnią z jasności wszystkich (nie zaliczają się do tego pikseli wyznaczone wcześniej przez maskę.) pikseli na odległości nie więcej 15 od aktualnie analizowanego.

Wynik finalny:



Rysunek 29: Przykładowe zdjęcie – porównanie efektu przed i po.

10. Uwagi co do działania programu

Przez to, że algorytm działa lokalnie (tylko w punktach wyznaczonych przez maskę) nie wpływa on na ogólny wygląd zdjęcia. Z tego też powodu możemy wykonywać program kilkakrotnie na zdjęciu – uzyskując lepsze efekty.

Końcowy algorytm składa się z pięciu iteracji opisanego wyżej algorytmu, znacząco zwiększając szansę na usunięcie zanieczyszczeń.

Ponadto można uzyskać dodatkowe informacje o działaniu programu i jakości zdjęcia – za przykład funkcja: `.countNoise()` zlicza ilość wyznaczonych przez maskę pikseli – które są uznane za zanieczyszczenia.

11. Dostępność programu

Na chwilę obecną nasze rozwiązanie jest programem terminalowym, działającym na systemie nie starszym niż Windows 10 – choć istnieje plan przeniesienia go także na inne popularne systemy operacyjne.

Tak samo pracujemy obecnie nad stworzeniem bardziej przystępnego interfejsu okienkowego.

Program dostępny jest na licencji *open source* i jego kod źródłowy można znaleźć na GitHubie pod adresem:

<https://github.com/ssk12o/PTI-Foto-Projekt>.

12. Wykorzystywane narzędzia

W tej części naszego projektu korzystaliśmy z następujących narzędzi:

- Programu i języka Matlab – do analizy zdjęć;
- Języka C++ – do napisania programu;
- Programu VS Code – do tworzenia, edycji i dokumentacji kodu programu i raportów;
- Programu LibreOffice Calc – do analizy części danych numerycznych;
- \LaTeX 2 ϵ – do przygotowania raportu;
- Strony Github i programu Git – do udostępniania, dystrybucji i pracy nad kodem;
- 7zip – do kompresji zdjęć;
- Google Drive – do udostępniania plików;
- Skanera minilab Noritsu HS-1800 – do wykonywania wysokiej jakości cyfrowych skanów zdjęć wykonanych techniką analogową;
- Aparatów:
 - Canon EOS 300 z obiektywem Tamron 28-105mm 1:4-5.6 i kliszą Fomapan 400
 - Fujifilm FinePix L55 Digital Camera – Black (12MP, 3x Optical Zoom)

13. Podział obowiązków

Na tym etapie projektu podzieliśmy się pracą, obowiązkami i zadaniami w następujący sposób:

- Bartosz Wójcik – wykonywanie, skanowanie i analiza zdjęć; opieka merytoryczna.
- Katarzyna Szwed – tworzenie, analizowanie i pisanie algorytmu; korekta raportu.
- Natalia Szymańska – pisanie raportu.
- Patrycja Szałajko – zarządzanie pracą zespołu, kontakt z mediami.
- Aleksandra Wójcik – skanowanie zdjęć rodzinnych w celu polepszenia ich jakości w końcowych etapach projektu.
- Karol Sęk – tworzenie, analizowanie i pisanie algorytmu.
- Michał Juskiewicz – tworzenie, analizowanie i pisanie algorytmu.
- Filip Sajko – pisanie raportu, implementacja w \LaTeX .