



Politechnika  
Śląska

**Politechnika Śląska**  
**Wydział Automatyki, Elektroniki i Informatyki**

Projekt inżynierski

Wykorzystanie sieci neuronowych do detekcji ziaren w obrazach  
termowizyjnych

Autor: Maciej Ziaja

Kierujący pracą: dr inż. Sebastian Budzan

Gliwice, styczeń 2020

## **Streszczenie**

Tematem pracy jest...

# Deklaracja

Deklaruję że...

# Podziękowania

Pragnę podziękować...

# Spis treści

<b>1</b>	<b>Wstęp</b>	<b>1</b>
1.1	Motywacja projektu . . . . .	1
1.2	Cel pracy . . . . .	1
<b>2</b>	<b>Założenia projektowe i wykorzystane narzędzia</b>	<b>2</b>
2.1	Analizowane pyły rud miedzi . . . . .	2
2.2	Rodzaje termowizji i idea wykorzystania pomiarów termowizyjnych	2
2.3	Kamera termowizyjna FLIR A320 . . . . .	3
2.3.1	Opis sprzętowy wykorzystywanej kamery . . . . .	3
2.3.2	Oprogramowanie do obsługi kamery . . . . .	5
2.4	Narzędzia programistyczne . . . . .	5
2.4.1	Język programowania Python . . . . .	5
2.4.2	Biblioteka przetwarzania obrazu Scikit-image . . . . .	6
2.4.3	Interfejs sieci neuronowych Keras i biblioteka TensorFlow . .	6
<b>3</b>	<b>Analiza i ekstrakcja danych z kamery termowizyjnej</b>	<b>7</b>
3.1	Proces pomiarowy i budowa zbioru danych . . . . .	7
3.2	Analiza zebranych obrazów termowizyjnych . . . . .	7
3.2.1	Prezentacja przykładowej serii pomiarowej . . . . .	7
3.2.2	Przetwarzanie danych wizyjnych . . . . .	7
3.2.3	Poprawa jakości obrazu . . . . .	7
3.2.4	Automatyczny odczyt zakresu pomiarowego temperatur z ob- razu . . . . .	7
3.3	Poszukiwanie zależności użytecznych w klasyfikacji . . . . .	8
3.3.1	Rozważane możliwości użycia cech obrazu w klasyfikacji . . .	8
3.3.2	Wybór algorytmu detekcji ziaren . . . . .	8
3.3.3	Algorytm śledzenia ziaren w serii zdjęć . . . . .	8
3.3.4	Wizualizacja zebranych cech i ocena ich użyteczności . . . .	8

<b>4</b>	<b>Prototypowanie sieci neuronowej klasyfikującej ziarna miedzi</b>	<b>10</b>
4.1	Budowa prototypu sieci neuronowej . . . . .	10
4.1.1	Dobór struktury sieci . . . . .	10
4.1.2	Trening sieci neuronowej . . . . .	10
4.2	Walidacja i ocena działania sieci . . . . .	10
<b>5</b>	<b>Podsumowanie</b>	<b>11</b>

# Rozdział 1

## Wstęp

### 1.1 Motywacja projektu

### 1.2 Cel pracy

## Rozdział 2

# Założenia projektowe i wykorzystane narzędzia

### 2.1 Analizowane pyły rud miedzi

### 2.2 Rodzaje termowizji i idea wykorzystania pomiarów termowizyjnych

Pomiary termowizyjne polegają na rejestracji promieniowania cieplnego obiektów w celu ustalenia ich temperatury. Badania z wykorzystaniem termowizji można podzielić na dwie grupy: *termowizję pasywną* oraz *termowizję aktywną*.

Technika pasywna polega na rejestracji obrazów obiektów bez ingerencji w ich temperaturę w czasie trwania pomiarów. Można w ten sposób obserwować przepływ ciepła w urządzeniach technicznych, procesach przemysłowych oraz biologicznych. W rozważanym w pracy przypadku technika ta nie ma jednak zastosowania. W warunkach pokojowych badany materiał ma w całej swojej objętości podobną temperaturę co skutkuje obrazem równomiernego szumu w rejestrowanym obrazie.

Bardziej zaawansowaną techniką są pomiary z wykorzystaniem termowizji aktywnej. W tym trybie pomiarów badane obiekty nagrzewa się w początkowej fazie pomiaru, w powtarzalnych warunkach, a następnie rejestruje proces ich stygnięcia. Podczas dostarczania ciepła do obiektu oraz jego oddawania na obrazach termowizyjnych można zaobserwować strukturę badanego obiektu. Przedmioty o niejednorodnej i złożonej budowie nagrzewają się oraz stygną nierównomiernie co jest rejestrowane przez kamery termowizyjne. W ten sposób można dostrzec cechy obiektu niewidoczne gołym okiem, takie jak zmiany jego gęstości, składu oraz uszkodzenia struktury. Termowizję aktywną stosuje się szeroko w badaniach naukowych oraz przemyśle. Przykładowe aplikacje tej techniki to wykrywanie defektów



w produktach przemysłowych oraz wykrywanie części konstrukcji podatnych na zużycie. Termowizja aktywna jest techniką niedestruktywną oraz bezkontaktową, co stanowi jej zalety w analizie materiałowej. Technika ta wymaga jednak etapowego procesu pomiarowego, potrzebne jest przygotowanie instalacji grzewczej, a nagrzewanie i stygnięcie materiału może być czasochłonne. Przebieg zmian temperatury najlepiej rejestrować na materiałach wideo, aby zmaksymalizować ilość danych zgromadzonych w trakcie eksperymentu.

## 2.3 Kamera termowizyjna FLIR A320

Przedmiotem projektu jest analiza obrazów pochodzących z przemysłowej kamery termowizyjnej FLIR A320. Jest to kamera łącząca wysoką jakość pomiaru z nowoczesnymi funkcjami integracji z oprogramowaniem komputerowym. Urządzenie łączy się z komputerem za pomocą kabla ethernetowego, pozwalając na kontrolę z poziomu oprogramowania oraz bibliotek programistycznych. Dodatkowo kamera posiada możliwość planowania automatycznych pomiarów, alarmów, oferuje funkcje analityczne oraz wbudowany serwer internetowy.

### 2.3.1 Opis sprzętowy wykorzystywanej kamery

Kamera ma postać podłużnego korpusu, do którego swobodnie można podłączać pasujące obiektywy. Na rysunku 2.1 przedstawiono zdjęcie wykorzystywanego urządzenia. Egzemplarz kamery znajdujący się w laboratorium termowizji Politechniki Śląskiej został wyposażony w obiektyw typu makro, pozwalający oglądać ziarna rud miedzi w bardzo dużym powiększeniu. Kamera cechuje się następującymi parametrami:

- typ detektora: niechłodzony mikrobolometr
- rozdzielczość: 320 na 240 pikseli,
- częstotliwość odświeżania: 9Hz do 30Hz
- szerokość otworu:  $f1.3$ ,
- autofokus z wbudowanym silnikiem,
- zakres pomiarowy temperatur:
  1. od  $-15^{\circ}\text{C}$  do  $50^{\circ}\text{C}$ ,
  2. od  $0^{\circ}\text{C}$  do  $350^{\circ}\text{C}$ ,
- dokładność:  $\pm 2^{\circ}\text{C}$  lub  $\pm 2\%$  odczytu,

- zakres wykrywanego widma promieniowania:  $7.5\mu\text{m}$  do  $13\mu\text{m}$ .



Rysunek 2.1: Kamera termowizyjna FLIR A320

Kamera wykrywa temperaturę przez detektor zwany *bolometrem*, który mierzy energię niesioną przez fale elektromagnetyczne w spektrum podczerwieni. Kiedy fala pada na detektor kamery, temperatura komórek matrycy rośnie i zwiększa się ich rezystancja elektryczna, co wpływa na natężenie prądu przepływającego przez obwody w czujniku. Wartości natężenia prądu są mierzone, a na ich podstawie określana jest mierzona temperatura.

Zakres temperatur kamery jest odpowiedni do przeprowadzenia eksperymentów z pomiarami metodą termowizji aktywnej. Próbkę planuje się podgrzewać do temperatury maksymalnie około  $80^{\circ}\text{C}$ , wartość ta mieści się w zakresie pracy urządzenia. Dokładność kamery jest zadowalająca, próbne materiały nagraniowe wskazały, że na zdjęciach widocznych jest wiele szczegółów i detali badanego materiału. Przy klasyfikacji obrazów i wzorców stygnięcia próbek jest to bardziej istotne niż liczbową dokładność pomiarowa przyrządu.

W porównaniu ze zwykłymi, współczesnymi aparatami rozdzielczość kamery termowizyjnej może wydawać się bardzo mała. Należy sobie jednak uzmysłwić, że w standardowych aparatach piksele mają rozkład Bayera, a wartości składowych koloru są interpolowane. W kamerze termowizyjnej każdy piksel dokonuje pełnego pomiaru wartości temperatury, dlatego bezpośrednie porównanie rozdzielczości używanego przyrządu z popularnymi aparatami może być mylące. Oczywiście większa rozdzielczość kamery byłaby pożądana, jednak jej obecne możliwości pozwalają na szczegółowe pomiary i obserwacje wielu detali ziaren rud miedzi.

### 2.3.2 Oprogramowanie do obsługi kamery

Jedną z najważniejszych cech kamery jest łatwość jej integracji z oprogramowaniem komputerowym. Kamere można obsługiwać za pomocą programu FLIR Tools. Pozwala on na podgląd obrazu z kamery oraz wykonywanie zdjęć termowizyjnych. Producent dostarcza również bibliotekę LabVIEW pozwalającą na zaawansowaną pracę z kamerą. Do obsługi stanowiska został napisany program używający tych bibliotek, który pozwala na nagrywanie materiałów wideo przy pomocy kamery. Nagrane materiały mają własnościowy format firmy FLIR, jednak można je odtwarzać w programie FLIR Tools. Program pozwala także na eksport stopklatek z nagrania, w postaci plików jpg. Przy obsłudze narzędzia ważne jest ustawianie zakresu temperatur na obrazie. Wybrany zakres decyduje w jaki sposób wartości temperatury są mapowane na kolory w obrazie zawarte w tablicy LUT. Program oferuje tablice w skali szarości, takie zostały użyte w projekcie, możliwy jest także wybór tablic w postaci kolorowych gradientów. Wybór zakresu wpływa na wygląd wyświetlanego obrazu oraz eksportowanych stopklatek. Jego nieodpowiedni dobór może skutkować zbyt ciemnym, jasnym, lub mało kontrastowym obrazem. Aby zapewnić najlepsze wykorzystanie nagrań z kamery oraz powtarzalny charakter eksportu stopklatek korzystano z opcji automatycznego doboru zakresu temperatur, jaki jest wbudowany w program FLIR Tools.

## 2.4 Narzędzia programistyczne

### 2.4.1 Język programowania Python

Założenia projektu wymagają użycia języka programowania pozwalającego na zaawansowaną obróbkę obrazu oraz wydajne budowanie sieci neuronowych. W obu tych dziedzinach wiodącym językiem jest Python, posiadający bogaty zestaw bibliotek. Język ten oferuje dużą wygodę programowania, co jest istotne przy prototypowaniu oraz pracy badawczej. Jednocześnie Python jest zaopatrzony w wydajną bibliotekę obliczeń numerycznych NumPy. Oferuje ona klasy macierzy numerycznych oraz bogaty zestaw operacji matematycznych. Macierze biblioteki NumPy są mniej elastyczne niż zwykłe listy języka Python, jednak oferują dużo większą wydajność obliczeniową, między innymi dzięki wsparciu obliczeń wektorowych na różnych procesorach. Wiele innych bibliotek wykorzystuje pakiet NumPy, na przykład przy przetwarzaniu obrazów, co pozwala na wysoką wydajność obliczeń. Połączenie wygody programowania z zadowalającą wydajnością bibliotek numerycznych, sprawia że język Python jest dobrym wyborem do realizacji założeń projektu.

### 2.4.2 Biblioteka przetwarzania obrazu Scikit-image

Język Python oferuje bogaty zestaw bibliotek przetwarzania obrazu. Najbardziej popularną z nich jest biblioteka OpenCV napisana w języku C++. Zdecydowano się jednak na wybór mniej znanego pakietu Scikit-image. Jest to biblioteka nastawiona bardziej na obliczenia naukowe i badawcze oraz napisana bezpośrednio z myślą o języku Python. Scikit-image, w porównaniu z OpenCV, oferuje bardziej spójny interfejs programisty oraz lepiej wykorzystuje charakterystykę języka Python. Dodatkowo wybrana biblioteka jest częścią zestawu Scikit, w skład którego wchodzi pakiet Scikit-learn służący do uczenia maszynowego. Może on być przydatny przy tworzeniu sieci neuronowej, a spójność bibliotek z pakietu Scikit jest niewątpliwie zaletą. Wybrana biblioteka charakteryzuje się również dobrą dokumentacją oraz zestawem przydatnych przykładów.

### 2.4.3 Interfejs sieci neuronowych Keras i biblioteka TensorFlow

Wybór bibliotek głębokiego uczenia w języku Python jest bardzo szeroki. Do najpopularniejszych pakietów należą: Keras, TensorFlow oraz PyTorch. Zdecydowano się na wybór interfejsu biblioteki Keras. Jest to pakiet nastawiony na elastyczność i możliwość eksperymentowania, oferuje wysokopoziomową, matematyczną warstwę abstrakcji opisu sieci neuronowych. Keras nie jest jednak pełnym rozwiązaniem, a raczej abstrakcyjnym interfejsem. Używanie go wymaga wyboru wewnętrznego silnika biblioteki. Zdecydowano się na wybór domyślnej opcji i użycie zaplecza pakietu TensorFlow.

## Rozdział 3

# Analiza i ekstrakcja danych z kamery termowizyjnej

### 3.1 Proces pomiarowy i budowa zbioru danych

### 3.2 Analiza zebranych obrazów termowizyjnych

#### 3.2.1 Prezentacja przykładowej serii pomiarowej

#### 3.2.2 Przetwarzanie danych wizyjnych

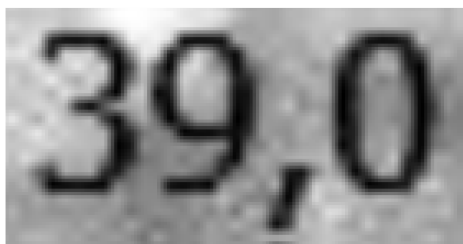
#### 3.2.3 Poprawa jakości obrazu

#### 3.2.4 Automatyczny odczyt zakresu pomiarowego temperatur z obrazu

Jak wspomniano w sekcji 2.3.2 jednym z kluczowych czynników decydujących o wyglądzie obrazów pochodzących z kamery jest zakres temperatur mapowany na kolory w obrazie. Niestety aplikacja FLIR Tools nie pozwala na eksport zakresu temperatur wraz ze zdjęciami w formie liczbowej. W czasie zapisu zdjęć oprogramowanie dodaje na nich interfejs z aktywną skalą pomiarową, jednak jest on graficznie naniesiony na obraz. Aby ułatwić w przyszłości pracę z materiałami z kamery opracowano dodatkowo mechanizm ekstrakcji zakresu temperatur z zdjęć pochodzących z kamery.

W celu konstrukcji funkcji odczytu wartości liczbowych z obrazu posłużono się gotową siecią neuronową zaprojektowaną do detekcji tekstu na zdjęciach. Zdecydowano się na użycie popularnej biblioteki *Pytesseract*. Aby poprawnie odczytać wartości z obrazu najpierw przycięto je tak by w kadrze znajdowała się tylko odczytywana liczba. Ponieważ przy eksporcie zdjęć program FLIR Tools nakłada inter-

fejs na zdjęcia w identyczny sposób, kadrowanie obrazu jest takie same dla każdej próbki pomiarowej. Wycięte kadry są bardzo małej rozdzielczości, aby ułatwić sieci rozpoznawanie liczb zdecydowano się przeskalować je w górę. W czasie skalowania włączono mechanizm anty aliasingu aby wyrównać krawędzie cyfr. Ponieważ używana sieć uznaje za tło kolor biały oraz poszukuje liczb w kolorze czarnym barwy na zdjęciu odwrócono. Następnie obraz poddano binaryzacji metodą *otsu*. Jest to popularna i wydajna metoda binaryzacji, jej efektywność jest maksymalna kiedy ilość pikseli tła oraz pierwszego planu jest zbliżona, dlatego poprawne kadrowanie liczb sprzyja jakości ich binaryzacji[1]. Na rysunku 3.1 przedstawiono kolejne etapy przygotowania obrazu do rozpoznania liczb. Implementację opisanego mechanizmu odczytywania zakresu temperatur ze zdjęć przedstawiono na listingu 1.



(a) Przeskalowany kadr z liczbą



(b) Kadr z liczbą po binaryzacji

Rysunek 3.1: Przygotowanie zakresu temperatur do odczytu przez sieć neuronową

### 3.3 Poszukiwanie zależności użytecznych w klasyfikacji

#### 3.3.1 Rozważane możliwości użycia cech obrazu w klasyfikacji

#### 3.3.2 Wybór algorytmu detekcji ziaren

#### 3.3.3 Algorytm śledzenia ziaren w serii zdjęć

#### 3.3.4 Wizualizacja zebranych cech i ocena ich użyteczności

```
def get_temperature_bounds(img, bounds=((6, 24), (283, 318)),  
                             ((219, 236), (283, 318))))):  
    '''Extract temperature values from FLIR UI on image.'''  
    img = invert(img)  
    temp_txt = []  
    for bound in bounds:  
        bound_img = img[slice(*bound[0]), slice(*bound[1])]  
        bound_img = rescale(bound_img, 4, anti_aliasing=True)  
        thr = threshold_otsu(bound_img)  
        img_txt = bound_img > thr  
        img_txt = Image.fromarray(img_txt)  
        temp = pytesseract.image_to_string(img_txt, config='digits')  
        if temp is not '':  
            temp = float(temp)  
        else:  
            temp = 0  
        temp_txt.append(temp)  
    return temp_txt
```

Listing 1: Funkcja języka Python do odczytywania zakresu temperatur ze zdjęć z kamery

## Rozdział 4

# Prototypowanie sieci neuronowej klasyfikującej ziarna miedzi

### 4.1 Budowa prototypu sieci neuronowej

#### 4.1.1 Dobór struktury sieci

#### 4.1.2 Trening sieci neuronowej

### 4.2 Walidacja i ocena działania sieci



## Rozdział 5

## Podsumowanie

# Spis rysunków

2.1	Kamera termowizyjna FLIR A320 . . . . .	4
3.1	Przygotowanie zakresu temperatur do odczytu przez sieć neuronową	8

## Spis tablic

# Spis listingów

1	Funkcja języka Python do odczytywania zakresu temperatur ze zdjęć z kamery . . . . .	9
---	--	---

# Bibliografia

- [1] Bülent Sankur Mehmet Sezgin. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13(1):146–165, 2004.