

**INSTYTUT ELEKTROTECHNIKI TEORETYCZNEJ**

**I SYSTEMÓW INFORMACYJNO-POMIAROWYCH**

Kierunek studiów – Elektrotechnika, II st.

Specjalność – Systemy wbudowane

Projektowanie systemów wizyjnych

**Analiza zmian skórnych z wykorzystaniem obrazu**

**Valentyna Dudnyk**

272831

**Przemysław Wiszniewski**

271006

Warszawa 2020

**1. Wprowadzenie**

Zadanie projektowe polegało na analizie zmian skórnych z wykorzystaniem obrazu za pomocą Pythona oraz OpenCV. Dodatkowo zostały wykorzystane biblioteki takie jak: numpy, scikit-learn, matplotlib, pandas. W ramach realizacji projektu został użyty zbiór do analizy pochodzący ze strony <https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DBW86T>. Podczas implementacji również wspomagano się informacjami zawartymi w artykule ze strony <https://www.nature.com/articles/sdata2018161>.

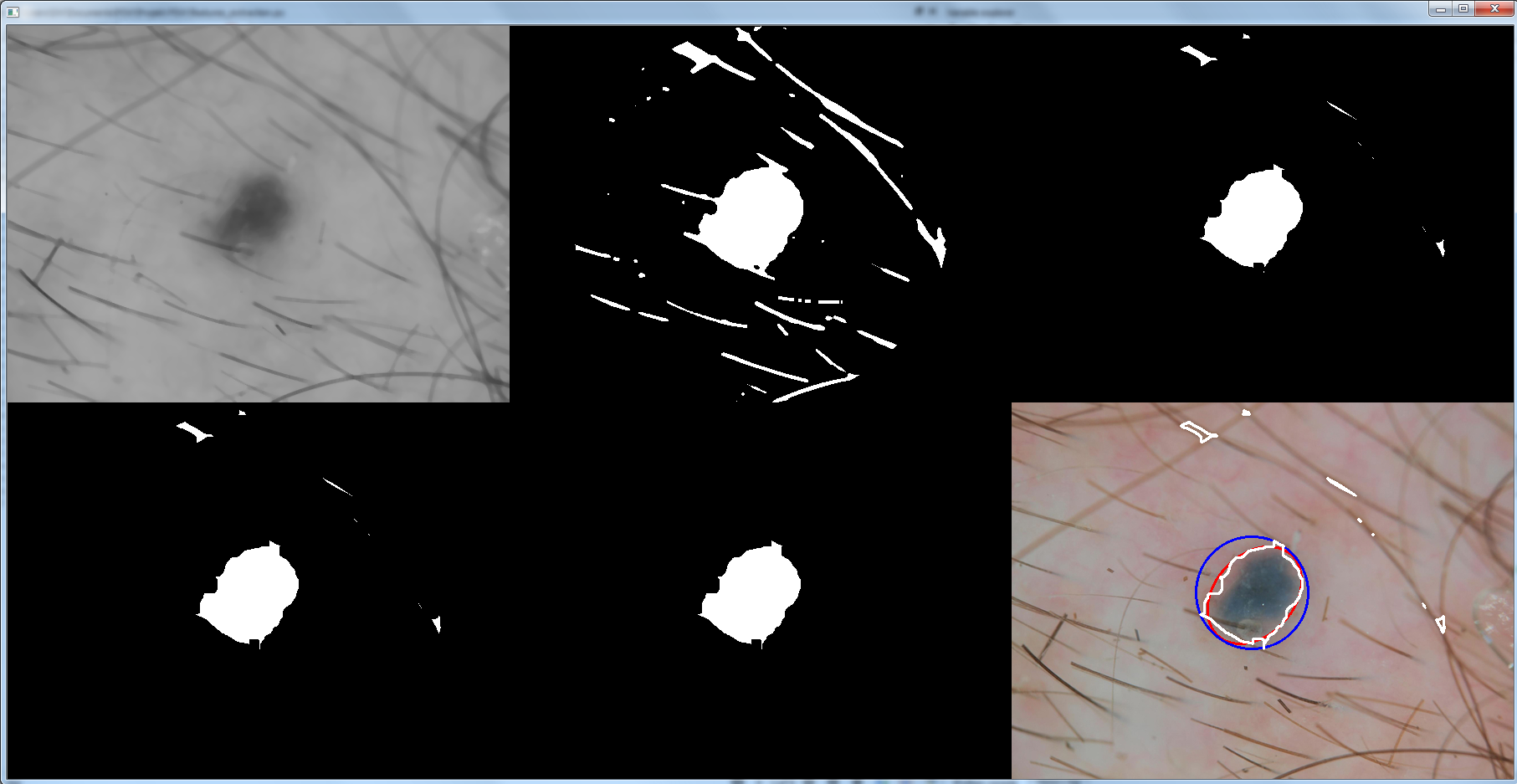
**3. Realizacja projektu**

Trzeba opisać zbiorem cech każde zdjęcie, a następnie porównać cechy z wykorzystaniem narzędzi statystycznych.

Projekt można podzielić na cztery zasadnicze części: selekcja obszaru, ekstrakcja cech charakteryzujących zmiany skórne, selekcja przydatnych cech oraz klasyfikacja.

**3.1 Selekcja obszaru**

Część pierwsza projektu polegała na znalezieniu i zaznaczeniu największego obszaru przenaczonego do dalszej analizy przy użyciu dostępnej bazy danych. Analizowany zbiór zawierał około 10000 zdjęć o 7 rodzajach zmian skórnych. Podczas tego etapu wykorzystano: medianBlur, threshold(THRESH\_OTSU), erose, close.



Rys.3.1 Proces obróbki obrazu

**3.2 Ekstrakcja cech**

Część druga projektu polegała na określeniu cech charakteryzujących zmiany skórne oraz implementację algorytmu obliczeniowego. W naszym projekcie na podstawie wyselekcjonowanego największego obszaru zostały wybrane 33 cechy: Hu Moments (hu), zajmowany obszar w pikselach (max\_area), iloraz zajmowanego obszaru do obszaru zajmowanego przez okrąg opisany na wybranym kształcie (circ\_area\_ratio), obwód zajmowanego obszaru (perimeter), iloraz półosi małej do półosi wielkiej elipsy opisanej na wybranym kształcie (min\_maj\_ell\_ratio), iloraz obwodu elipsy do obwodu obszaru (perimeter\_ratio) oraz iloraz obwodu do powierzchni zajmowanej. Zostały obliczone również średnia wartość, wariancja, skośność i kurtoza wystąpujących w wyselekcjonowanym obszarze kolorów RGB (red, green, blue), odcieni szarości (gray) oraz odcień światła (hue) z przestrzeni barw HSV (mean, var, skew, kurt).

1 hu0

2 hu1

3 hu2

4 hu3

5 hu4

6 hu5

7 hu6

8 max\_area

9 circ\_area\_ratio

10 perimeter

11 min\_maj\_ell\_ratio

12 perimeter\_ratio

13 mean\_blue

14 mean\_green

15 mean\_red

16 mean\_gray

17 mean\_hue

18 var\_blue

19 var\_green

20 var\_red

21 var\_gray

22 var\_hue

23 skew\_blue

24 skew\_green

25 skew\_red

26 skew\_gray

27 skew\_hue

28 kurt\_blue

29 kurt\_green

30 kurt\_red

31 kurt\_gray

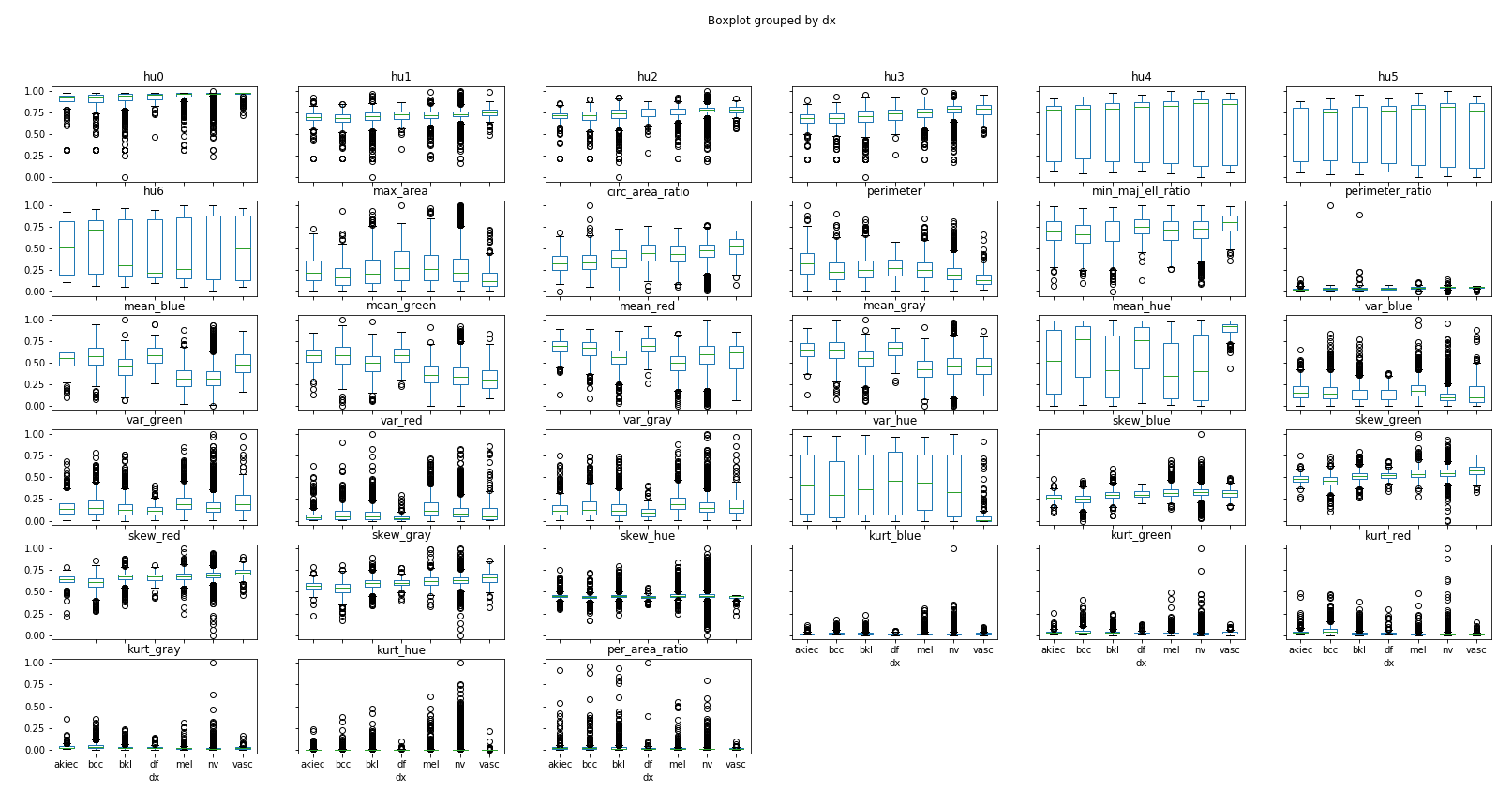
32 kurt\_hue

33 per\_area\_ratio

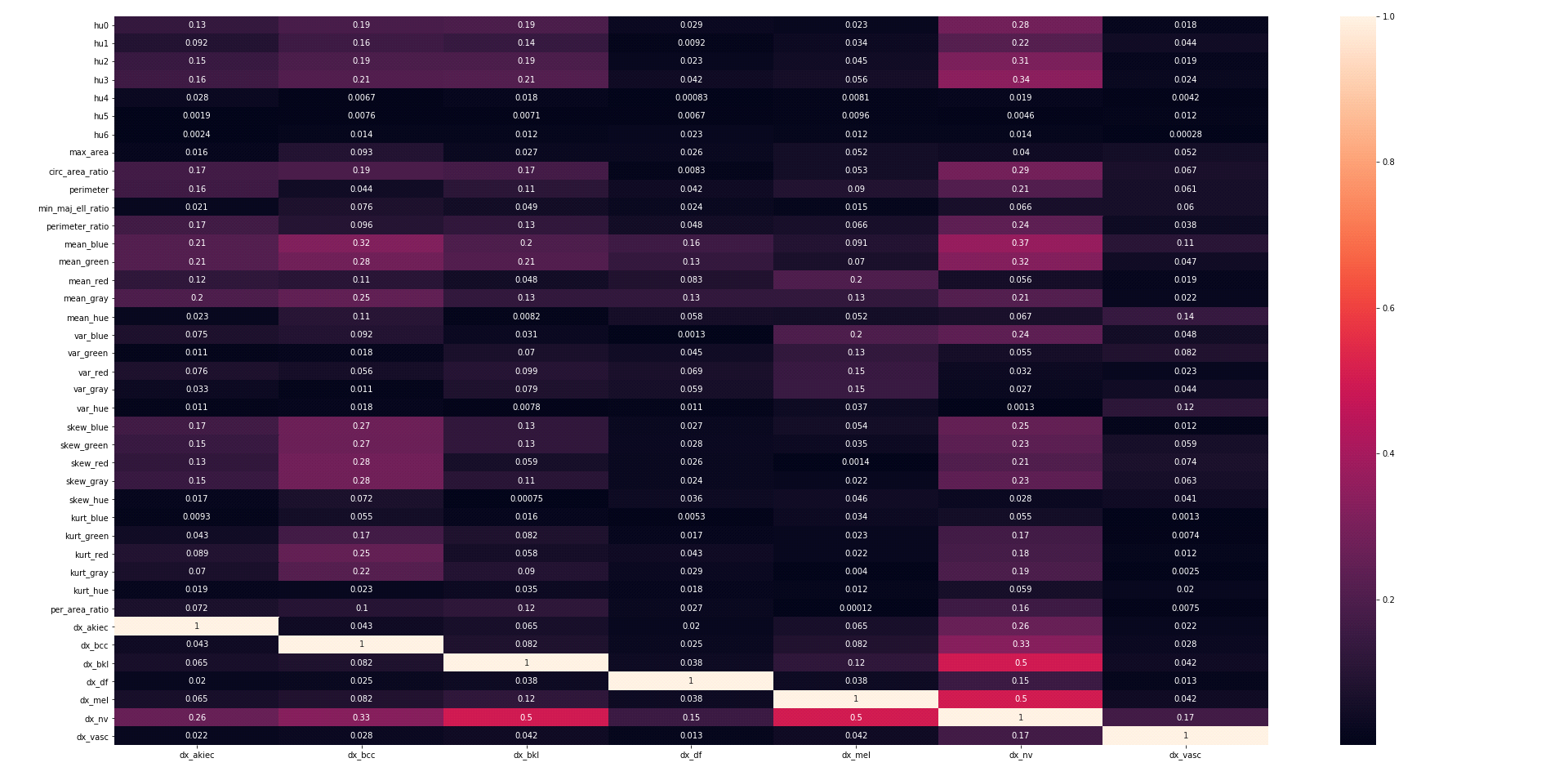
**3.3 Selekcja cech**

Wstępnie selekcja została dokonana za pomocą narzędzi boxplot z biblioteki pandas.

Sprawdzono również korelację poszczególnych cech z rodzajami zmian skórnych (dx\_akiec, dx\_bcc, dx\_bkl, dx\_df, dx\_mel, dx\_nv, dx\_vasc).



Rys.3.3.1 Wykresy pudełkowe



Rys.3.3.2 Macierz korelacji cech oraz klas

Analizując uzyskane wyniki zostało wyselekcjonowanych 20 cech: 'hu2', 'hu3', 'max\_area', 'circ\_area\_ratio', 'perimeter', 'min\_maj\_ell\_ratio',

'mean\_blue', 'mean\_green', 'mean\_red', 'mean\_gray', 'mean\_hue',

'var\_blue', 'var\_green', 'var\_red', 'var\_gray', 'var\_hue', 'skew\_blue',

'skew\_green', 'skew\_red', 'skew\_gray'.

Ponadto do wysekecjonowania cech wykorzystano algorytm SelectKBest z biblioteki scikit-learn. W tym przypadku zadana liczba cech do wyboru również wynosiła 20.

'hu0', 'hu1', 'hu2', 'hu3', 'circ\_area\_ratio', 'perimeter', 'mean\_blue',

'mean\_green', 'mean\_red', 'mean\_gray', 'mean\_hue', 'var\_blue',

'var\_green', 'var\_red', 'var\_gray', 'var\_hue', 'skew\_blue',

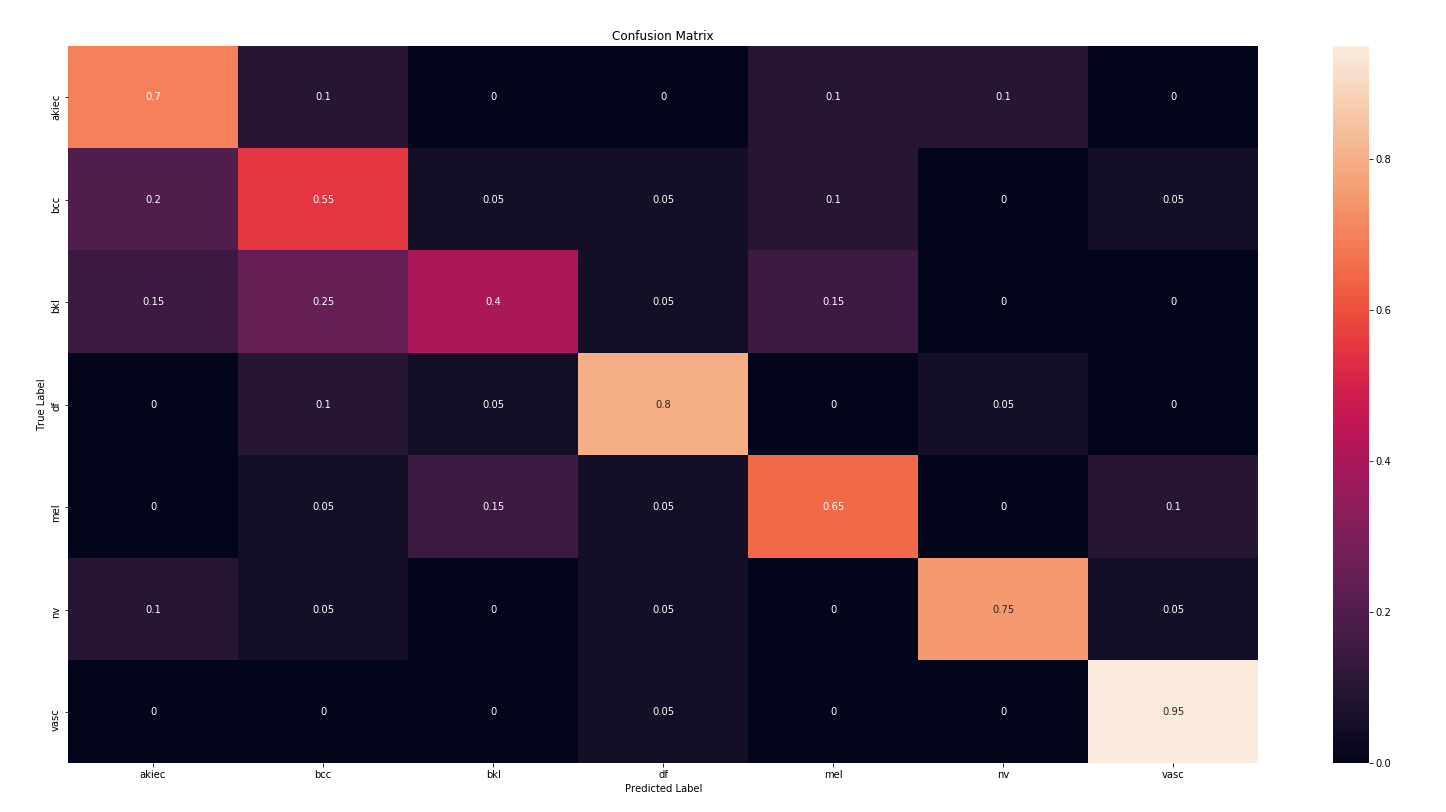
'skew\_green', 'skew\_red', 'skew\_gray'.

**3.4 Klasyfikacja oraz wyniki badań (testów)**

Na tym etapie projektu podzielono dane na zbiory treningowy (80%) i testowy (20%). Następnie użyto klasyfikatora SVM. Przeprowadzono poszukiwanie najlepszych parametrów modelu używając algorytmu GridSearchCV z 5-krotną kroswalidacją uzyskując best\_params = {'C': 4.4, 'class\_weight': 'balanced', 'gamma': 'scale', 'kernel': 'rbf'}.

My:

dla 100:



dx

akiec 20

bcc 20

bkl 20

df 20

mel 20

nv 20

vasc 20

Name: dx, dtype: int64

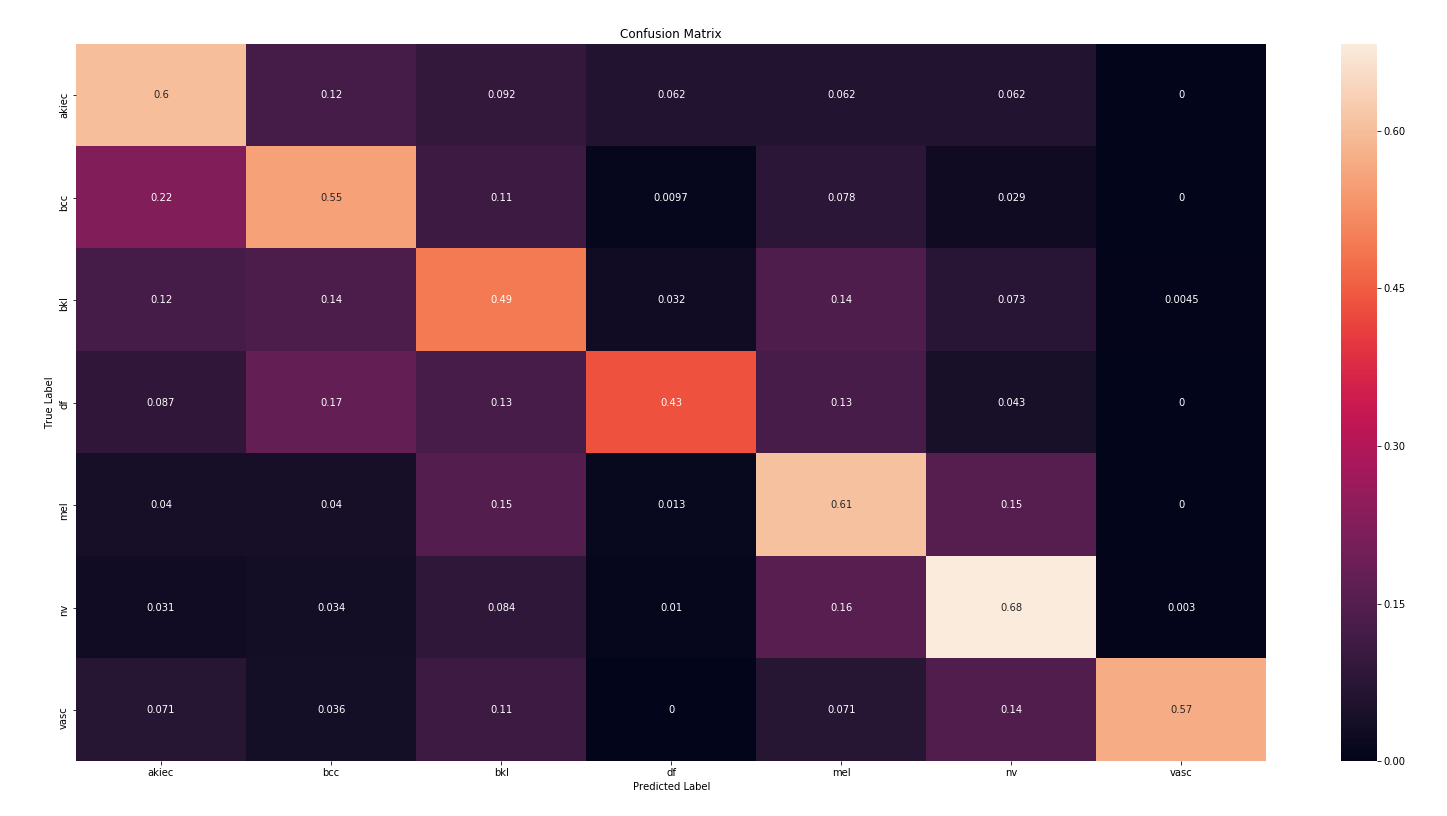
Accuracy: 0.69

Precision: 0.69 [0.61 0.5 0.62 0.76 0.65 0.83 0.83]

Recall: 0.69 [0.7 0.55 0.4 0.8 0.65 0.75 0.95]

F1\_score: 0.68 [0.65 0.52 0.48 0.78 0.65 0.79 0.88]

dla wszy



dx

akiec 65

bcc 103

bkl 220

df 23

mel 223

nv 1341

vasc 28

Name: dx, dtype: int64

Accuracy: 0.64

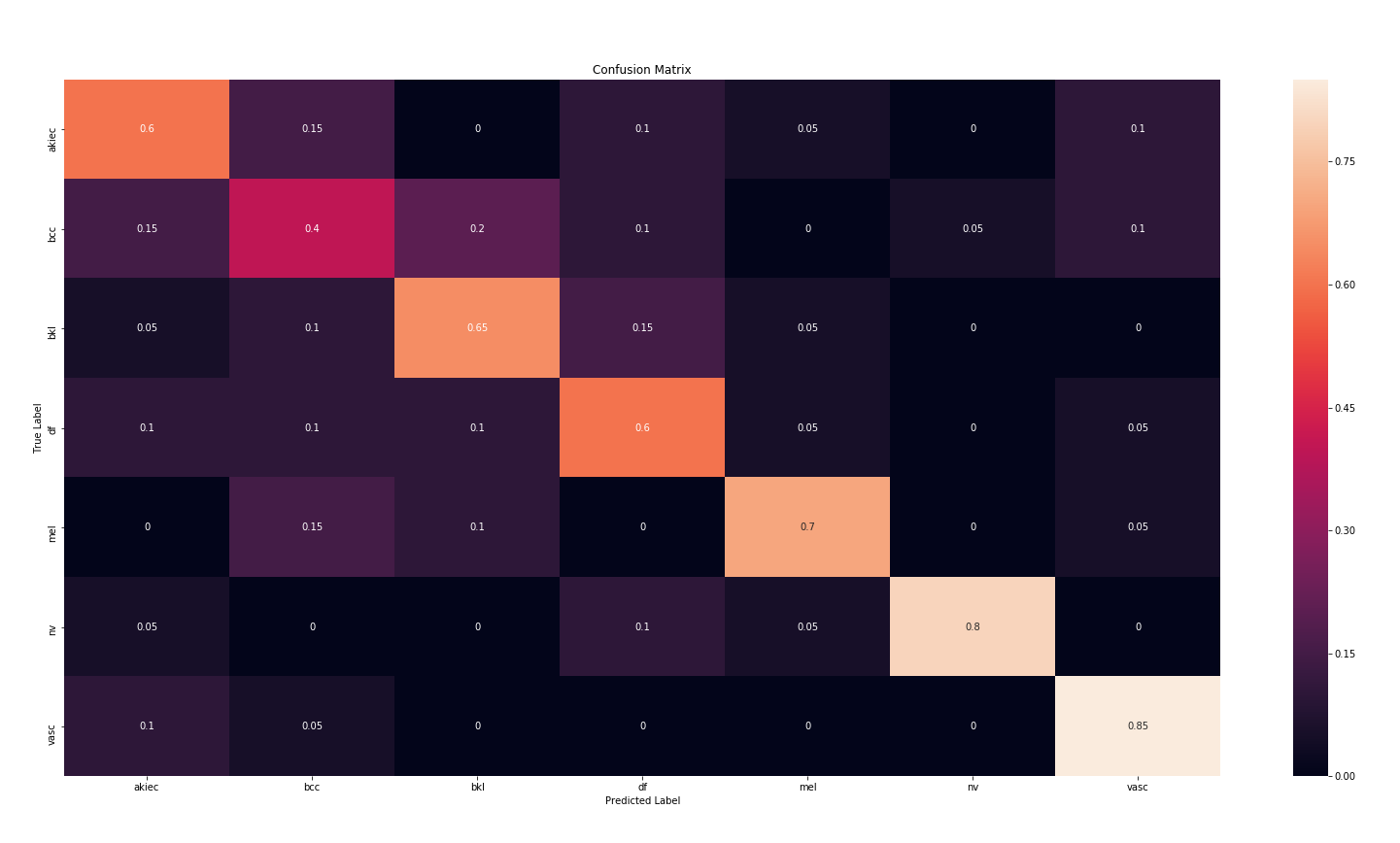
Precision: 0.48 [0.27 0.37 0.39 0.26 0.35 0.94 0.76]

Recall: 0.56 [0.6 0.55 0.49 0.43 0.61 0.68 0.57]

F1\_score: 0.49 [0.37 0.44 0.43 0.32 0.44 0.79 0.65]

----------------funkcja

dla100:



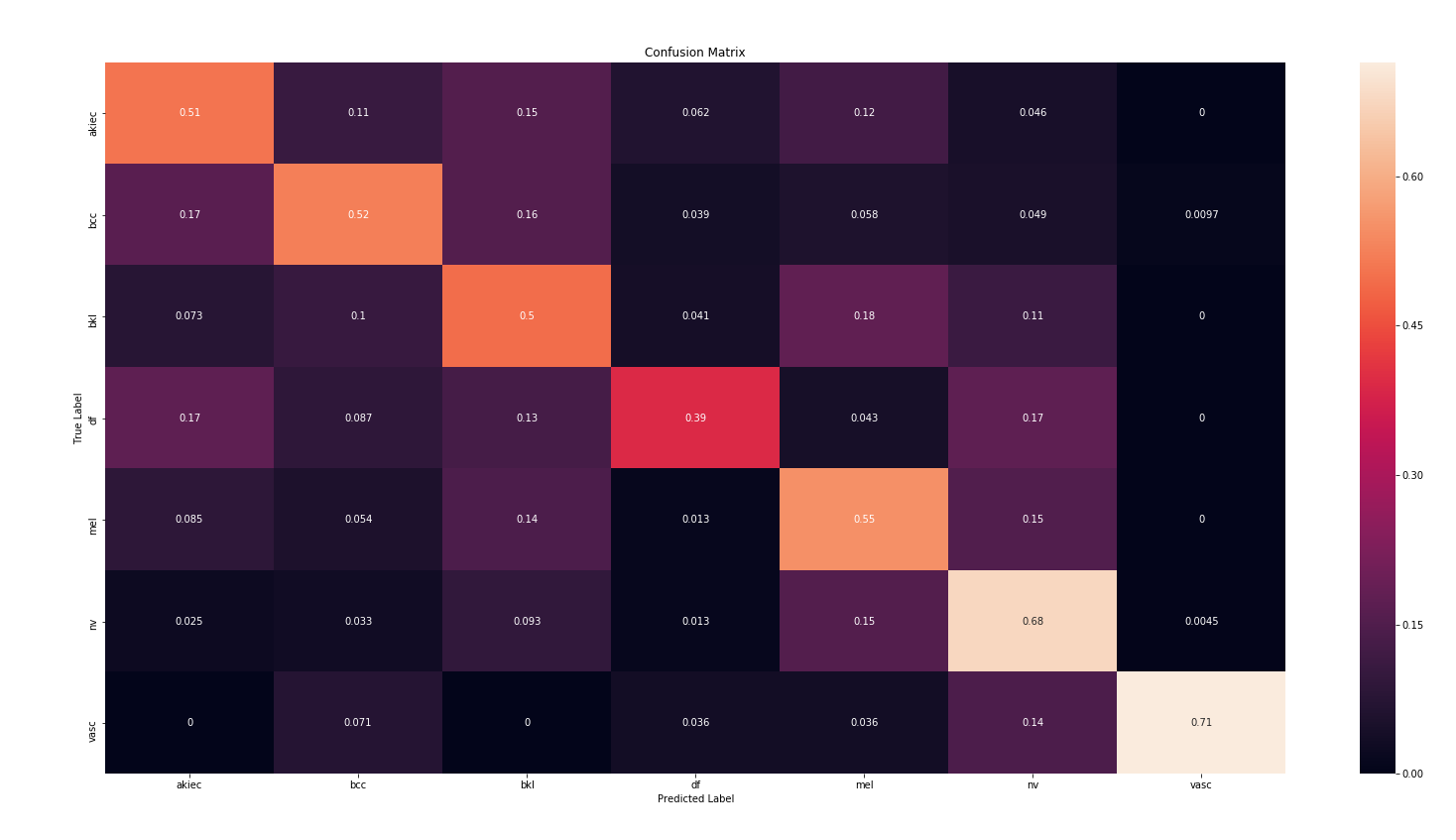
Accuracy: 0.66

Precision: 0.66 [0.57 0.42 0.62 0.57 0.78 0.94 0.74]

Recall: 0.66 [0.6 0.4 0.65 0.6 0.7 0.8 0.85]

F1\_score: 0.66 [0.59 0.41 0.63 0.59 0.74 0.86 0.79]

dla wszystkich



dx

akiec 65

bcc 103

bkl 220

df 23

mel 223

nv 1341

vasc 28

Name: dx, dtype: int64

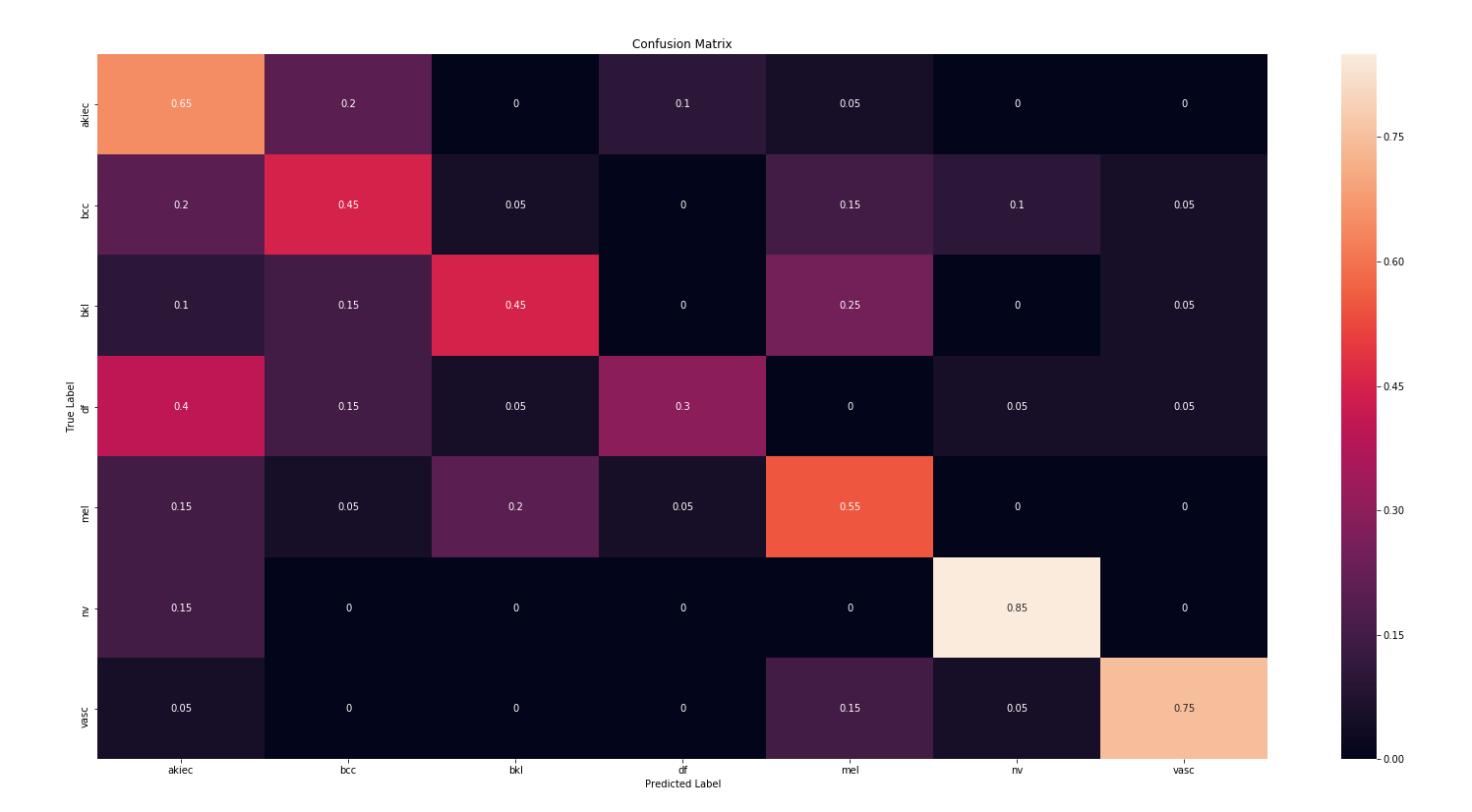
Accuracy: 0.63

Precision: 0.46 [0.27 0.38 0.37 0.19 0.32 0.92 0.74]

Recall: 0.55 [0.51 0.52 0.5 0.39 0.55 0.68 0.71]

F1\_score: 0.48 [0.35 0.44 0.42 0.26 0.4 0.78 0.73]

gorsze wyniki dla 5 cech i 100:



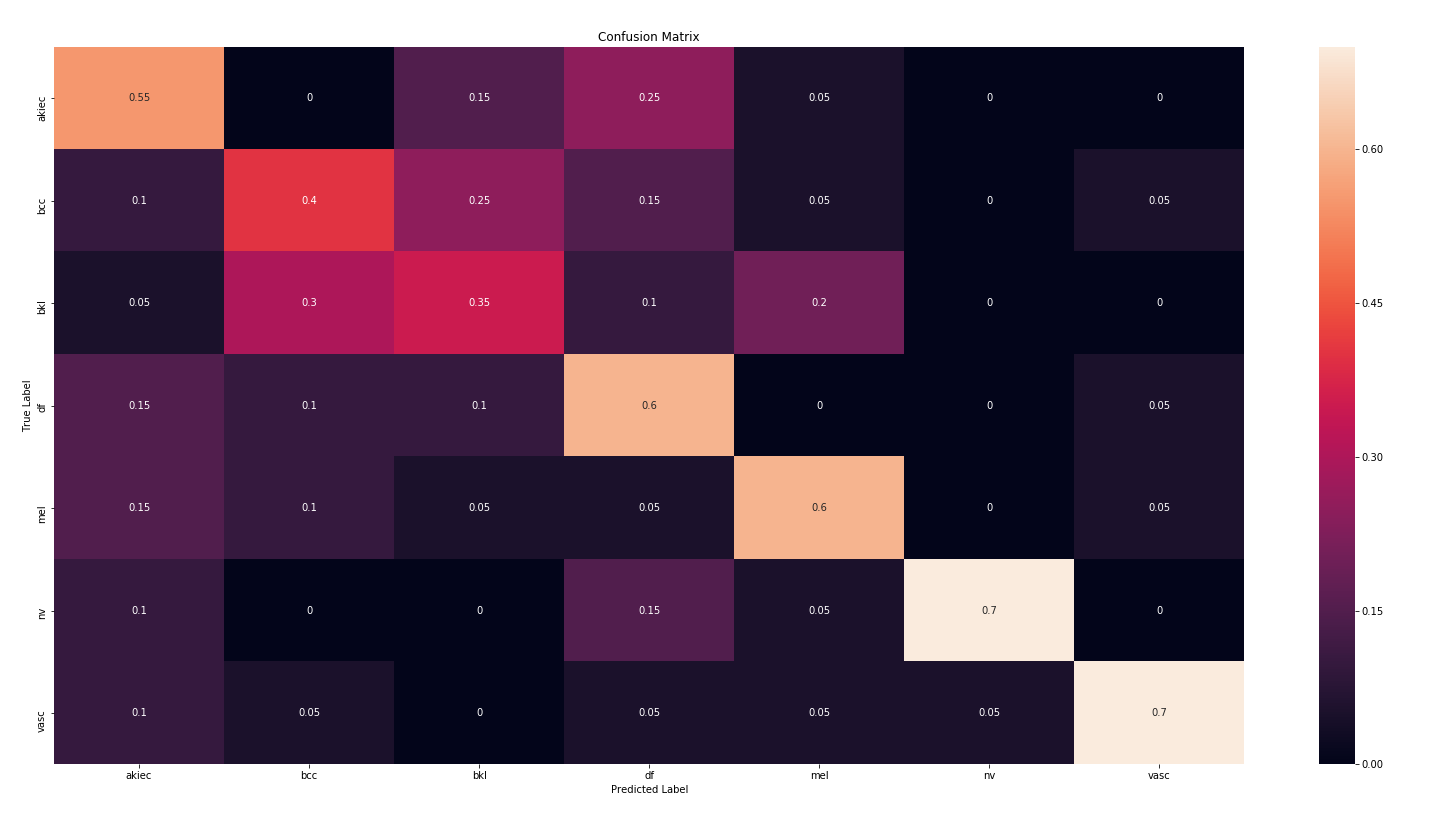
Accuracy: 0.55

Precision: 0.55 [0.43 0.23 0.54 0.57 0.4 0.85 0.83]

Recall: 0.55 [0.5 0.15 0.35 0.65 0.6 0.85 0.75]

F1\_score: 0.54 [0.47 0.18 0.42 0.6 0.48 0.85 0.79]

wszystkie 33 cechy i 100:



Accuracy: 0.56

Precision: 0.58 [0.46 0.42 0.39 0.44 0.6 0.93 0.82]

Recall: 0.56 [0.55 0.4 0.35 0.6 0.6 0.7 0.7 ]

F1\_score: 0.56 [0.5 0.41 0.37 0.51 0.6 0.8 0.76]

**zakres trafności 58-71**

**4. Podsumowanie i wnioski końcowe**

