

Uniwersytet Warszawski
Wydział Fizyki

Urszula Romaniuk

Nr albumu: 347442

**Wytwarzanie sztucznych obrazów
uczących na potrzeby uczenia
maszynowego**

Praca licencjacka
na kierunku Zastosowania Fizyki w Biologii i Medycynie
specjalność Neuroinformatyka

Praca wykonana pod kierunkiem
dr. Józefa Gintera
Zakład Fizyki Biomedycznej
Instytut Fizyki Doświadczalnej

Warszawa, Wrzesień 2017

Oświadczenie kierującego pracą

Oświadczam, że niniejsza praca została przygotowana pod moim kierunkiem i stwierdzam, że spełnia ona warunki do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

Data

Podpis kierującego pracą

Oświadczenie autora (autorów) pracy

Świadom odpowiedzialności prawnej oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie i nie zawiera treści uzyskanych w sposób niezgodny z obowiązującymi przepisami.

Oświadczam również, że przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem tytułu zawodowego w wyższej uczelni.

Oświadczam ponadto, że niniejsza wersja pracy jest identyczna z załączoną wersją elektroniczną.

Data

Podpis autora (autorów) pracy

Streszczenie

Celem pracy było zbadanie skuteczności metod sztucznego powiększania zbioru uczącego obrazów na potrzeby uczenia maszynowego. W pracy użyto zdjęć skórnych zmian barwnikowych złośliwych i niezłośliwych pobranych z International Skin Imaging Collaboration. Zastosowano bibliotekę Pythona Image, dzięki której wytworzono sztuczne obrazy uczące. Obrazy zostały sparametryzowane za pomocą wyuczonych sieci konwolucyjnych, udostępnionych przez BVLC. Model VGG-CNN-S pochodzi z Image Large Scale Visual Recognition Challenge i został nauczony na bazie danych z ImageNet. Następnie do sparametryzowanych obrazów zastosowano klasyfikator wektorów nośnych SVM. Sprawdzono zbiory uczące zrównoważone (zawierające w przybliżeniu równe liczby obrazów z kategorii zmian barwnikowych złośliwych i niezłośliwych) oraz zbiory niezrównoważone (o różnych liczbach obrazów z wymienionych wcześniej kategorii). Testy zostały przeprowadzone na zbiorach zrównoważonych i niezrównoważonych. Wyniki sugerują, że zarówno do uczenia klasyfikatora SVM jak i jego testowania lepiej używać zbiorów zrównoważonych.

Słowa kluczowe

data augmentation, sztuczne obrazy uczące, machine learning, uczenie maszynowe, klasyfikator SVM, sieci konwolucyjne, czerniak, skórna zmiana barwnikowa

Dziedzina pracy (kody wg programu Socrates-Erasmus)

13.2 Fizyka

Tytuł pracy w języku angielskim

Using data augmentation for the purposes of machine learning

Spis treści

| | |
|---|----|
| Cel pracy | 3 |
| 1. Wstęp | 4 |
| 1.1. Sztuczne sieci neuronowe | 5 |
| 1.2. Konwolucyjne sieci neuronowe | 7 |
| 1.3. Maszyny wektorów nośnych,(ang. Support vector machine) - SVM | 9 |
| 1.4. Różnicowanie zmian barwnikowych łagodnych i złośliwych w praktyce klinicznej | 11 |
| 1.5. Wytwarzanie sztucznych obrazów uczących | 12 |
| 1.6. Sieci konwolucyjne z maszyną wektorów nośnych - CSVM | 13 |
| 1.7. Metody oceny klasyfikatora | 13 |
| 2. Dane eksperymentalne | 15 |
| 3. Metodologia | 17 |
| 3.1. Oprogramowanie | 17 |
| 3.2. Klasyfikacja zdjęć. Metoda pracy | 17 |
| 4. Wyniki | 20 |
| 4.1. Dane treningowe budowane na zbiorze niezrównoważonym | 20 |
| 4.2. Dane treningowe budowane na zbiorze zrównoważonym | 22 |
| 4.3. Krzywa ROC | 25 |
| 4.4. Wnioski | 25 |
| 5. Podsumowanie | 28 |

Cel pracy

Celem pracy było porównanie metod sztucznego zwiększania zbioru obrazów uczących wykorzystanych do klasyfikacji metodą SVM w oparciu o parametryzację z konwolucyjnej sieci neuronowej na przykładzie zdjęć skórnych zmian barwnikowych niezłośliwych i złośliwych (czerniaka).

Rozdział 1

Wstęp

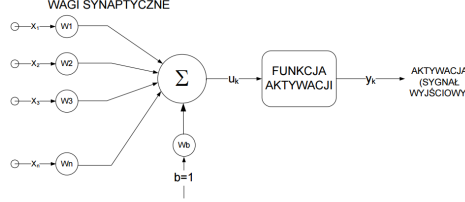
Prawie każdy człowiek posiada na swoim ciele zmiany barwnikowe, które od lat nie zmieniają swojego kształtu i barwy. Zdecydowana większość z nich ma charakter łagodny i ich występowanie nie jest objawem żadnej choroby, jednakże niewielka część zmienia swój wygląd na przestrzeni tygodni, co może świadczyć o ich złośliwym charakterze. Taką zmianą jest czerniak, czyli nowotwór złośliwy wywodzący się z komórek barwnikowych skóry - melanocytów, który ma postać zmiany barwnikowej na skórze lub błonie śluzowej i obecnie stanowi on poważny problem zdrowia publicznego. Czynnikiem predysponującym do zachorowania na ten nowotwór jest nadmierna ekspozycja na promieniowanie UV, co dotyczy w szczególności osób o jasnej karnacji. Jego częstość występowania wzrasta co roku i on jest jednym z najbardziej powszechnych nowotworów złośliwych w populacji całego świata. Niestety sama obserwacja ewolucji zmiany, bez oglądania jej w znacznym powiększeniu zwiększa ryzyko nacieku miejscowego głębokich warstwy skóry oraz przerzutów do odległych narządów, co może czynić z czerniaka chorobę trudno wyleczalną z potencjalnie śmiertelnym skutkiem. Im mniejszy jest nowotwór i im płycej nacieka skórę, tym więcej pacjentów przeżywa 5 lat od rozpoznania czerniaka. Dlatego wczesna diagnoza i jak najszybsze usunięcie potencjalnie złośliwej zmiany barwnikowej pozwala zapobiec przerzutom i zwiększa szanse na wyleczenie nawet do 90%. Wczesne wykrycie, skuteczne metody obrazowania i wdrożenie odpowiedniego leczenia onkologicznego może być bardzo skuteczne w leczeniu czerniaka [1, 2].

Dermatoskopia polega na oglądaniu skóry człowieka w powiększeniu przy pomocy dermatoskopu, co ułatwia identyfikację zmian skórnych. Jest to o tyle lepsza metoda od oglądania zmiany "gołym" okiem, że umożliwia wizualizację jej struktur powierzchniowych i podpowierzchniowych. Badanie dermatoskopowe wykonywane przez doświadczonych specjalistów ma nawet 90% czułości i 56% swoistości. Niestety u mniej doświadczonych lekarzy współczynnik ten spada do 62-63% [3]. Głównym problemem tego badania jest to, że diagnoza jest subiektywną oceną opartą na algorytmach takich jak reguła ABCDE, reguła trzypunktowa, reguła siedmiopunktowa, metoda Menziesa, algorytm CASH, które oparte są morfologicznych szczegółach zmiany barwnikowej [4]. Dermatoskopia przyczyniła się do rozwoju komputerowych technik wizualizacji. Na Międzynarodowym Sympozjum do Spraw Obrazowania Biomedycznego (ISBI - International Symposium on Biomedical Imaging) zorganizowano konkurs przy współpracy z International Skin Imaging Collaboration (ISIC). Celem tego wyzwania było wspieranie badań i opracowanie algorytmów do komputerowego rozpoznawania czerniaka przy użyciu 900 obrazów uzyskanych w dermatoskopie.

W pracy użyto właśnie tego zestawu zdjęć do przeprowadzenia badania nad sztucznie powiększonym zbiorem uczącym na potrzeby uczenia maszynowego.

1.1. Sztuczne sieci neuronowe

Sztuczne warstwowe sieci neuronowe (SSN) składają się z neuronów uporządkowanych w warstwy. Wyróżnia się warstwę wejściową, warstwę ukrytą i warstwę wyjściową. Na Ry-



Rysunek 1.1: Na rysunku przedstawiona jest ogólna zasada działania pojedynczego neuronu. W neuronie sumowany jest wektor wejściowy z odpowiadającymi mu wagami. Następnie suma ta jest argumentem dla funkcji aktywacji. Po zastosowaniu wszystkich operacji tworzone jest wyjście neuronu.[5].

sunku 1.1 przedstawiona jest zasada działania pojedynczego neuronu. Neuron można traktować jako przetwornik sygnału, gdzie na wejściu znajduje się wektor danych wejściowych $x = (x_1, x_2, \dots, x_{n-1}, x_n)$. Wektory te pochodzą z parametryzacji danych treningowych, na których uczą się sieci. Neuron z warstwy ukrytej jest połączony z wszystkimi neuronami z poprzedniej warstwy. Każde połączenie między neuronami scharakteryzowane jest za pomocą wag w . Sumaryczne wejście neuronu u_k przedstawia wzór 1.1. W neuronie sumowane są dane wejściowe wraz z odpowiadającymi im wagami.

$$u_k = \sum_{j=1}^n w_{kj} \cdot x_j + w_b \quad (1.1)$$

gdzie k to indeks przypisany neuronowi w warstwie. Dodatkowa waga w_b - wyrazu wolnego wprowadzana jest, aby uzyskać wyraz niezależny. w_b można zapisać jako wagę z indeksem zero w_0 . W tym wypadku do wektora wejściowego przyjmuje się sztucznie dodatkową cechę x_0 zawsze równą 0. Wtedy wzór (1.1) można zapisać w postaci iloczynów skalarnych (1.2).

$$u_k = \sum_{j=0}^n w_{kj} \cdot x_j = w_k^T x \quad (1.2)$$

$$u = W \times x \quad (1.3)$$

gdzie W to macierz o wymiarach $k \times n$ wag, a x to wektor o długości n cech. u to wektor o długości k odpowiadający ilości klas, a n to liczba cech na wejściu.

Aktywacja neuronu a_k to funkcja aktywacji zastosowana do sumy danych wejściowych wraz z odpowiadającymi im wagami(1.4).

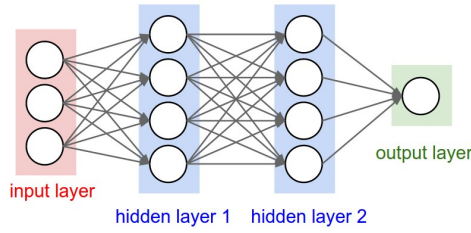
$$a_k = f(u_k) \quad (1.4)$$

Przykładowe funkcje aktywacji to:

- funkcja skokowa:

$$f(u, t) = \begin{cases} 0 & u < t \\ 1 & u > t \end{cases} \quad (1.5)$$

gdzie t - wartość progowa.



Rysunek 1.2: Na rysunku przedstawiona jest ogólna zasada sieci jednokierunkowych, która składa się z warstwy wejściowej (kolor czerwony), dwóch warstw ukrytych (kolor niebieski) i warstwy wyjściowej (kolor zielony) [6].

- funkcja logistyczna:

$$f(u) = \frac{1}{1 + \exp^{-u}} \quad (1.6)$$

- tangens hiperboliczny:

$$f(u) = \frac{\exp^u - \exp^{-u}}{\exp^u + \exp^{-u}} \quad (1.7)$$

- funkcja prostownicza:

$$f(u) = \max(0, c) \quad (1.8)$$

gdzie $c = Wx + b$

Funkcja skokowa przyjmuje tylko dwie wartości i jest nieróżniczkowalna w t . Funkcja logistyczna zawiera się w przedziale $(0, 1)$, a tangens hiperboliczny w przedziale $(-1, 1)$. Obie są różniczkowalne w całej przestrzeni. Funkcja prostownicza przyjmuje wartości $\langle 0, \infty \rangle$, a pochodną tej funkcji jest funkcja skokowa, gdzie $t = 0$.

Wyróżnia się kilka rodzajów sieci neuronowych w zależności od typu budowy, organizacji i postaci połączeń między neuronami. Jednym z rodzajów są sieci jednokierunkowe, przez które sygnał przechodzi tylko raz i nie zachodzi w nich sprzężenie zwrotne. Sieci rekurencyjne to takie, w których dopuszczalne są w nich sprzężenia zwrotne, które polegają na tym, że na wejście w danej warstwie podawany jest sygnał wyjściowy z innego neuronu z tej samej warstwy. W niektórych przypadkach mogą występować sprzężenia bezpośrednie, w których sygnał wyjściowy wchodzi ponownie do tego samego neuronu na wejściu. W przypadku całkowicie połączonych ze sobą sieci mogą występować wszystkie rodzaje połączeń między neuronami, bez połączeń bezpośrednich. W przypadku klasyfikatorów binarnych warstwa wyjściowa składa się z jednego neuronu, który zwraca liczbę z przedziału $(0, 1)$. Aby uzyskać predykcję klasy 0 lub 1, używa się funkcję skokową z ustalonym progiem t . W przypadku klasyfikatorów, które posiadają więcej niż dwie klasy, warstwa wyjściowa składa się z wektora o długości K , gdzie K to ilość klas. W tego rodzaju klasyfikatorach predykcję systemu stanowi indeks odpowiadający maksymalnej wartości w wektorze wyjściowym. Numer warstwy $l = 1, 2, 3, \dots, L - 1, L$, gdzie L - liczba warstw. Zatem aktywację z warstwy l stanowi wejście w warstwie $l + 1$. Schemat budowy sieci jednokierunkowej przedstawiono na rysunku 1.2

Proces uczenia maszynowego polega na zachodzącym w trakcie uczenia poprawianiu i dobraniu odpowiednich wag tak, aby sygnał przechodzący przez sieci neuronowe dał prawidłową predykcję na wyjściu. W tym celu stosuje się propagację wsteczną, która polega na tym, że przy losowych wagach sygnał najpierw propaguje się do przodu, po czym propaguje się do tyłu. W tym procesie zwracana jest informacja o błędzie do wag, w których on wystąpił, proporcjonalnie do jego wielkości. Na tej podstawie aktualizowana jest wartość wag. Celem

uczenia maszynowego jest takie dobranie wag, aby zminimalizować średni błąd kwadratowy (funkcja kosztu) popelniany na ciągu uczącym. Jeśli dany jest zbiór uczący postaci

$$\left\{ \left(X^{(j)}, z^{(j)} \right) \right\}_{j=1,2,\dots,N-1,N}$$

Funkcja błędu przyjmuje postać

$$J(w) = \frac{1}{2} \sum_{j=1}^N \left(z^{(j)} - u^{(j)} \right)^2 = \sum_{j=1}^N J^{(j)}(w)$$

gdzie $u^{(j)} = f(e^{(j)}) = f\left(\sum_{i=0}^n w_i^{(j)} x_i^{(j)}\right)$ zaś $J^{(j)}(w) = \frac{1}{2} \left(z^{(j)} - y^{(j)} \right)^2$. Następny krok jest zmiana wag w kierunku przeciwnym do kierunku gradientu funkcji kosztu. i -ta waga zmienia się w następujący sposób (1.9).

$$w_i^{(j+1)} - w_i^{(j)} = \Delta w_i^{(j)} = -\eta \frac{\partial J^{(j)}}{\partial w_i} = -\eta \frac{\partial J^{(j)}}{\partial u^{(j)}} \frac{\partial u^{(j)}}{\partial w_i} = -\eta \frac{\partial J^{(j)}}{\partial u^{(j)}} \frac{\partial u^{(j)}}{\partial e} \frac{\partial e}{\partial w_i} \quad (1.9)$$

Zauważając, że

$$\begin{aligned} \frac{\partial J^{(j)}}{\partial u^{(j)}} &= -\left(z^{(j)} - u^{(j)} \right) \\ \frac{\partial e}{\partial w_i} &= x_i^{(j)} \\ \frac{\partial u^{(j)}}{\partial e^{(j)}} &= \frac{df(e)}{de^{(j)}} \end{aligned}$$

Stosując te przekształcenia otrzymano (1.10).

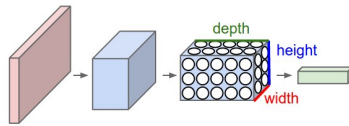
$$\Delta w_i^{(j)} = \eta \frac{df(e)}{de^{(j)}} \left(z^{(j)} - y^{(j)} \right) x_i^{(j)} = \eta \delta^{(j)} x_i^{(j)} \quad (1.10)$$

gdzie wprowadzono wzór na błąd (1.11)

$$\delta^{(j)} = \frac{df(e)}{de^{(j)}} \left(z^{(j)} - u^{(j)} \right) \quad (1.11)$$

1.2. Konwolucyjne sieci neuronowe

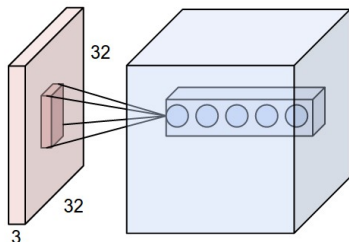
Konwolucyjne sieci neuronowe w swojej budowie przypominają sztuczne sieci neuronowe opisane w powyższym podrozdziale. Składają się z połączonych ze sobą neuronów, które tworzą zazwyczaj sieci jednokierunkowe. Proces propagacji sygnału w sieci konwolucyjnej polega na tym, że do kolejnego neuronu podawany jest sygnał z poprzedniego, z odpowiednimi wagami, które można modyfikować na drodze uczenia się. Neuron w każdej warstwie jest połączony z niewielką liczbą neuronów z warstwy poprzedniej. Oblicza on swoje pobudzenie jako iloczyn skalarny wektora wejść i wektora wag, a następnie przekształca je nieliniowo za pomocą funkcji aktywacji. Sieci konwolucyjne są konstruowane z założeniem, że na wejściu podawany jest obraz. Neurony sieci konwolucyjnej są zorganizowane w 3-wymiarową strukturę: szerokość, wysokość i głębokość. Przetwarzanie danych wejściowych polega na zmniejszeniu ich szerokości i wysokości a zwiększeniu grubości, co ostatecznie prowadzi do wytworzenia ostatniej



Rysunek 1.3: Na wejściu podany jest obraz jako 3-wymiarowa macierz mająca swoją szerokość, wysokość i grubość. Grubość ma rozmiar 3 odpowiadającą 3 kanałom RGB. Każda warstwa przekształca macierz wejściową na macierz wyjściową, w taki sposób, że zmniejszeniu ulega szerokość i wysokość macierzy, a zwiększeniu ulega grubość. Ostatecznie na wyjściu powstaje wektor o rozmiarach $1 \times 1 \times k$, gdzie k to liczba klas [6].

warstwy o wymiarach $1 \times 1 \times k$, gdzie k odpowiada liczbie klas. Ogólna zasada działania sieci konwolucyjnych przedstawiona jest na Rysunku 1.3.

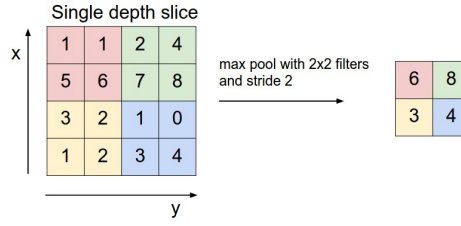
Podstawowymi rodzajami warstw w sieciach konwolucyjnych są: warstwa konwolucyjna, warstwa zwężająca oraz warstwa w pełni połączona. Warstwa konwolucyjna składa się z wielu map neuronów (identycznych w ramach każdej mapy), których macierz wag wejściowych można potraktować jako filtr przestrzenny; jego parametry ustalane są w procesie uczenia sieci. Taki filtr zorganizowany jest w 3-wymiarową macierz, która ma szerokość, wysokość i głębokość. Każdy filtr przesuwa się po wymiarach przestrzennych objętości. W każdej pozycji filtra obliczany jest spłot (iloczyn skalarny) tego filtra z lokalnym fragmentem objętości wejściowej. Jeden filtr produkuje na wyjściu macierz 2D odpowiadającą swojej aktywacji dla wszystkich położów w objętości wejściowej. Mapy aktywacji od pojedynczych filtrów składane są w objętości w 3-wymiarową macierz poprzez ułożenie ich w stos, wzdłuż wymiaru głębokości. Głębokość odpowiada ilości filtrów użytych w tej warstwie. Sposób działania sieci konwolucyjnej opisano na Rysunku 1.4.



Rysunek 1.4: Na rysunku przedstawiona jest zasada działania warstwy konwolucyjnej, gdzie w danym fragmencie obrazu następuje jego spłot z filtrem obrazu. Im więcej filtrów zastosowano, tym grubsza macierz powstaje na wyjściu[6].

W warstwie zwężającej zachodzi zmniejszenie liczby parametrów w sieci. Zwężanie polega na zmniejszeniu szerokości i wysokości macierzy. Głębokość nie ulega zmianie. Najczęściej stosowane są operacje, które biorą maksymalną wartość z fragmentu danej warstwy. Innym sposobem zwężania warstwy jest obliczenie sumy lub średniej wszystkich wartości z fragmentu warstwy. Warstwa zwężająca stosowana jest do zmniejszenia nadmiernego dopasowania, rysunek 1.5

Warstwa w pełni połączona jest zbudowana tak samo, jak w standardowej sieci neuronowej. Każdy neuron jest połączony ze wszystkimi neuronami z warstwy poprzedniej z odpowiadającymi im wagami. Dodatkowo w sieciach konwolucyjnych stosowane są warstwy, które wprowadzają nieliniowość. Najczęściej stosowane są warstwy RELU (ang. rectified linear unit), czyli funkcja prostownicza (1.8), ale można także stosować inne funkcje. Warstwy RELU występują zaraz po warstwie konwolucyjnej.



Rysunek 1.5: Na rysunku przedstawiona jest zasada działania warstwy zwężającej (ang.pooling). Proces ten polega na wzięciu maksymalnej wartości z części obrazu, dzięki czemu zmniejsza się jego szerokość i wysokość [6].

Konwolucyjna sieć neuronowa składa się z kilku (N) par konwolucyjno - prostowniczych, pomiędzy którymi może znaleźć się warstwa zwężająca. Po kilku takich blokach (M) występuje warstwa w pełni połączona w ilości od jednej do kilku (K). Te warstwy wykonują końcową predykcję. Na końcu takiej sieci znajduje się warstwa długości równa liczbie klas. Można to opisać poniższym schematem:

$$WEJCIE \longrightarrow [[CONV \longrightarrow RELU] * N \longrightarrow POOL?] * M \longrightarrow [FC \longrightarrow RELU] * K \longrightarrow FC$$

gdzie *CONV* to warstwa konwolucyjna, *RELU* to funkcja aktywacji, funkcja prostownicza, *POOL* to warstwa zwężająca, a *FC* to warstwa w pełni połączona. N,M,K to ilość wystąpień a ? oznacza może, ale nie musi wystąpić. W pracy wykorzystano uprzednio wyuczone na zbiorze niemedycznym konwolucyjne sieci neuronowe. Sieci neuronowe potraktowano jako "czarną skrzynkę", do której włożono obrazy i uzyskano obraz sparametryzowany na cechy. Służyły do tego warstwy w pełni połączone przed uzyskaniem końcowej predykcji.

1.3. Maszyny wektorów nośnych,(ang. Support vector machine) - SVM

Maszyny wektorów nośnych (z ang. Support Vector Machines,SVM) są popularnym algorytmem stosowanym w uczeniu maszynowym jako klasyfikator. Nauka klasyfikatora ma na celu wyznaczenie hiperpłaszczyzny rozdzielającej dane należące do dwóch różnych klas, maksymalizując marginesy rozdzielające dane położone najbliżej płaszczyzny. SVM ma tę zaletę, że sprawdza się również wtedy, gdy zbiór uczący składa się z mniejszej ilości obrazów, niż liczba cech zastosowana do parametryzacji każdego z obrazów.

Klasyfikator SVM jest klasyfikatorem binarnym, który na wejściu przyjmuje parę (x, y) , gdzie x to n-wymiarowy wektor cech, a y to prawidłowa klasyfikacja (0 lub 1). Klasyfikator ten jest opisany przez wektor parametrów $\Theta = (\theta_1, \theta_2, \dots, \theta_{n-1}, \theta_n)$ i dodatkowy składnik regulujący θ_0 . W klasyfikatorze SVM, jak w każdym algorytmie uczenia z nadzorem, celem optymalizacji jest minimalizacja funkcji kosztu. Funkcja, która zamienia wektor na skalar, tzw. funkcja decyzji, to iloczyn skalarny wektora wag i wektora wejść wraz z parametrem θ_0 .

$$U = f(x) = \Theta^T x + \theta_0 \quad (1.12)$$

Funkcja kosztu przyjmuje postać:

$$J(\Theta) = C \sum_{i=1}^m \left(y^{(i)} H_1(u^{(i)}) + (1 - y^{(i)}) H_0(u^{(i)}) \right) + \frac{1}{2} \sum_{j=1}^n \theta_j^2 \quad (1.13)$$

Suma $\frac{1}{2} \sum_{j=1}^n \theta_j^2$ to parametr regularyzacji. Dzięki tej sumie niwelowany jest problem wysokiej wariancji oraz przetrenowania, na co duży wpływ ma wysoka wartość θ_j , który obniża wartość parametru regularyzacji oraz zmniejsza ryzyko wystąpienia przetrenowania. Współczynnik C jest wagą składnika regularyzacji. Im mniejsze C , tym większe znaczenie ma regularyzacja. Funkcje H_0 i H_1 to funkcje losowe, które przyjmują postać:

$$H_0 = \max(0, z + 1) \quad (1.14)$$

$$H_1 = \max(0, 1 - z) \quad (1.15)$$

Z równań wynika, że aby uzyskać optymalnie dobrane parametry należy spełnić warunki:

$$\begin{aligned} \Theta^T x + \theta_0 &\geq 1 & \text{gdy } y^{(i)} &= 1 \\ \Theta^T x + \theta_0 &\leq -1 & \text{gdy } y^{(i)} &= 0 \end{aligned}$$

Przy wysokiej wartości C algorytm będzie dążył do tego, by zostało spełnione założenie dla wszystkich $i(1, 2, \dots, n-1, n)$. Składnik $\sum_{j=1}^n \theta_j^2$ można zapisać w innej formie jako kwadrat długości wektora Θ w n -wymiarowej przestrzeni.

$$\sum_{j=1}^n \theta_j^2 = \|\Theta\|^2 \quad (1.16)$$

Gdy powyższy warunek jest spełniony wtedy pierwszy składnik $J(\Theta)$ równa się 0. Cel optymalizacji możemy potraktować jako minimalizację długości wektora Θ , co można zapisać w postaci:

$$\min_{\Theta} \left(C \sum_{i=1}^m \left(y^{(i)} H_1(\Theta^T x^{(i)} + \theta_0) + (1 - y^{(i)}) H_0(\Theta^T x^{(i)} + \theta_0) \right) + \frac{1}{2} \|\Theta\|^2 \right) \quad (1.17)$$

Klasyfikator SVM posiada także geometryczną interpretację. Każdy element $x^{(i)}$ to punkt w n -wymiarowej przestrzeni, gdzie n to liczba cech. Θ definiuje hiperpłaszczyznę, która dzieli przestrzeń na obszary odpowiadające klasom. W zależności od tego, gdzie pojawi się nowy, nieznany wcześniej klasyfikatorowi punkt, zostanie on przydzielony do odpowiadającej im klasy. Taka hiperpowierzchnia określa się jako powierzchnię decyzji. Punkty, które leżą najbliżej tej powierzchni, zwane są wektorami nośnymi, a suma odległości tych wektorów od powierzchni separującej to margines. W przypadku interpretacji geometrycznej, należy tak dobrać parametry Θ , aby zmaksymalizować marginesy.

Niektóre punkty różnią się od pozostałych mimo, że należą do tej samej klasy. Są to punkty odstające. Wysoki parametr C pilnuje, aby te punkty należały do odpowiedniej klasy i przy tworzeniu hiperpowierzchni algorytm będzie je uwzględniał. W przypadku niskiego współczynnika C algorytm będzie ignorował takie punkty, co prowadzi do generalizacji hiperpowierzchni.

Wszystkie opisane powyżej przypadki odseparowane są hiperpłaszczyzną. Często jednak dane należy odseparować bardziej złożonymi funkcjami o nieliniowej powierzchni. W takich przypadkach należy przekształcić dane do innej przestrzeni. Do tego służy funkcja przekształcająca Φ :

$$\begin{aligned} \Phi : \mathbb{R}^n &\rightarrow \mathbb{R}^{n'} \\ x &\rightarrow \Phi(x) \\ (x_1, x_2, \dots, x_{n-1}, x_n) &\rightarrow (z_1, z_2, \dots, z_{n'-1}, z_{n'}) \end{aligned}$$

gdzie n' - wymiar nowej przestrzeni z_j - nowe cechy. Funkcja decyzji przyjmuje postać:

$$f(x) = f(\Phi(x)) = \Theta^T \Phi(x) + \theta_0 \quad (1.18)$$

gdzie: $\Theta \in \mathbb{R}^{n'}$. Kiedy n' jest bardzo duże, Θ zajmuje bardzo dużo pamięci. W takim przypadku stosuje się teorię reprezentacji [7, 8], która w przypadku SVM przyjmuje postać:

$$\Theta = \sum_{i=1}^m \alpha_i \Phi(x^{(i)}) \quad (1.19)$$

dla danych wartości α_i . Więc zamiast optymalizować Θ optymalizuje się α . Stąd funkcja decyzji:

$$f(x) = \sum_{i=1}^m \alpha_i \Phi(x^{(i)}) \Phi(x) + \theta_0 \quad (1.20)$$

Funkcja

$$K(x^{(i)}, x) = \Phi(x^{(i)}) \Phi(x) \quad (1.21)$$

jest zwana funkcją kernelową lub kernelem. Wtedy cel optymalizacji z danym kernelem przyjmuje postać:

$$\min_{\Theta, \theta_0} \left(C \sum_{j=1}^m \left(y^{(j)} H_1(f(x)) + (1 - y^{(j)}) H_0(f(x)) \right) + \frac{1}{2} \left\| \sum_{i=1}^m \alpha_i \Phi(x^{(i)}) \right\|^2 \right) \quad (1.22)$$

gdzie:

$$f(x) = \sum_{i=1}^m K(x^{(i)}, x) + \theta_0 \quad (1.23)$$

Za odwzorowanie danych w nowej przestrzeni odpowiedzialna jest funkcja kernelowa, która może przyjmować następujące postaci:

- Kernel wielomianowy:

$$K(x, x') = (\gamma * x * x' + 1)^d \quad \text{gdzie } \gamma > 0$$

- Kernel RBF:

$$K(x, x') = \exp(-\gamma \|x - x'\|^2)$$

- Kernel minimum:

$$K(x, x') = \sum_{j=1}^n \min(x_j, x'_j)$$

gdzie γ i d są parametrami jądrowymi. W niniejszej pracy został zastosowany kernel RBF.

1.4. Różnicowanie zmian barwnikowych łagodnych i złośliwych w praktyce klinicznej

Różnicowanie zmian barwnikowych jest rutynowym zadaniem lekarzy dermatologów. Aby zwiększyć wiarygodność badania używają oni dermatoskopu, który powiększa obraz widziany przez nich gołym okiem, co zwiększa trafność diagnozy. Różnice morfologiczne między zmianami barwnikowymi łagodnymi i złośliwymi są ujęte w skalach ocen. Do cech w nich uwzględnianych należą między innymi:

- asymetria zmiany barwnikowej
- atypowa siatka barwnikowa, inaczej nieregularne rozmieszczenie barwnika na zmianie barwnikowej
- niebiesko-biały welon
- atypowe naczynia, inaczej nieregularny układ naczyń krwionośnych
- nieregularne pasma - fachowe nazwy na rozmieszczenie barwnika
- nieregularne kropki/globule
- nieregularne plamy
- pola zaniku
- obecność 3 lub więcej kolorów
- nieregularne brzegi zmian skórnych
- zmiana wielkości i kształtu w przeciągu 3 miesięcy od poprzedniego badania

Lekarz na podstawie badania dermatoskopowego i oceny w jednej ze skal (7-punktowa lista Argenziano, metoda Menziesa, reguła ABCD) może tylko podejrzewać czerniaka i zalecić jak najszybsze całkowite usunięcie zmiany. Rozpoznanie stawiane jest na podstawie badania histopatologicznego wyciętej zmiany [4]. Więcej informacji na temat czerniaków [2].

1.5. Wytwarzanie sztucznych obrazów uczących

Wytwarzanie sztucznych obrazów uczących stosuje się na potrzeby poprawienia wydajności klasyfikatora oraz zmniejszenia przetrenowania. Te dwa zjawiska mogą wynikać z małej ilości danych użytych do wytrenowania odpowiedniego klasyfikatora, przez co algorytm nie jest w stanie dobrze rozdzielić danych na klasy. Zwiększając dane treningowe za pomocą różnych transformacji poprawie ulega finałowa trafność klasyfikatora. W niniejszej pracy zastosowano algorytm, w którym najpierw obrazy zostają przepuszczone przez wytrenowane sieci konwolucyjne, gdzie w wyniku tej operacji powstają obrazy sparametryzowane na cechy, a następnie tak sparametryzowany obraz zostaje przepuszczony przez klasyfikator SVM, gdzie zwracana jest predykcja. Dane treningowe można powiększyć na dwa sposoby. Albo poprzez transformacje obrazów przed przepuszczeniem obrazów przez sieci konwolucyjne, albo poprzez transformacje wytworzone na już sparametryzowanych cechach obrazu. W pracy [9] przetestowano wpływ transformacji obrazów na dwóch sposobach powiększania zestawu uczącego. Wyniki uzyskane dla klasyfikatora SVM wskazywały na to, że transformacja sparametryzowanych cech obrazu nie poprawiła wyników klasyfikatora. W związku z tym, w niniejszej pracy skupiono się tylko na transformacji obrazów uczących przed przepuszczeniem obrazu przez sieci konwolucyjne. Z powodu ograniczonej liczby danych, które można użyć do wytrenowania klasyfikatora, a zwłaszcza obrazów medycznych z każdej dziedziny medycyny, stosuje się sztuczne wytwarzanie danych uczących (ang. Data augmentation). Wytwarzanie sztucznych danych uczących polega na wytworzeniu nowych obrazów z jednego oryginalnego. Często w literaturze dotyczącej uczenia rozpoznawania czerniaka przez klasyfikator, z użyciem sieci konwolucyjnych [10] opisywane jest stosowanie transformacji geometrycznych i kolorystycznych. Transformacje geometryczne polegają na odbiciu obrazu wzdłuż poziomej i pionowej

osi obrazu, wycinaniu jego środka z zachowaniem krawędzi zmiany skórnej, a także rotacji obrazu o kąt 0° , 90° , 180° , 270° . Z kolei transformacje kolorystyczne zachodzą poprzez losową zmianę wartości w kanałach RGB (odpowiednio czerwony, zielony, niebieski). Stosowane jest także losowe przycinanie obrazu, wycinanie losowych fragmentów obrazu, przyciemnianie obwodu obrazu i losowe rotacje obrazu od prostej powierzchni obrazu [11, 12, 13]. W większości przypadków zwiększenie zbiorów treningowych prowadziło do zlikwidowania przetrenowania, a zmiany dokonane na klasyfikatorze poprawiły jego trafność.

1.6. Sieci konwolucyjne z maszyną wektorów nośnych - CSVM

W niniejszej pracy sieci konwolucyjne zostały użyte do parametryzacji cech obrazów, a klasyfikator maszyny wektorów nośnych do rozdzielania danych na odpowiadające im klasy. Połączenie tych dwóch klasyfikatorów jest częstym zabiegiem w operowaniu na danych testowych w postaci obrazów. SVM pozwala na uczenie za pomocą zbioru mniej liczniejszego od liczby parametrów parametrów cech obrazu. Wyniki uzyskiwane tą metodą dają niski poziom błędu, a klasyfikator jest wydajny [14, 15].

1.7. Metody oceny klasyfikatora

W uczeniu maszynowym z nadzorem stosuje się podział danych na treningowe i testowe. W procesie uczenia się klasyfikator pobiera dane z zestawu treningowego. Dla każdego elementu z zestawu pobierana jest para $(X^{(i)}, y^{(i)})$, gdzie $X^{(i)}$ to wektor cech, a $y^{(i)}$ to prawidłowa klasa. Następnie zostaje wykonana predykcja na przykładach z zestawu testowego, w której algorytm sam przydziela dane do prawidłowej dla niego klasy i porównuje ją z elementem ze zbioru testowego, gdzie zapisana jest rzeczywista, prawidłowa klasa.

Efektywność uczenia maszynowego zazwyczaj oceniana jest przez macierz błędów (Tabela 1.1

| | klasa predykowana prawdziwa | klasa predykowana nieprawdziwa |
|--------------------------------|-----------------------------|--------------------------------|
| klasa rzeczywista prawdziwa | prawdziwie pozytywna PP | fałszywie negatywna FN |
| klasa rzeczywista nieprawdziwa | fałszywie pozytywna FP | prawdziwie negatywna PN |

Tabela 1.1: Macierz błędów.

Danym przydzielane są odpowiednie etykiety pozytywne i negatywne. Jeśli dane przydzielone są jako pozytywne i są w rzeczywistości pozytywne, to przypisywane są do tabeli prawdziwie pozytywne (PP), jeśli nie są w rzeczywistości prawdziwe to umieszczane są w rubryce fałszywie pozytywne (FP). W przypadku gdy dane zostaną zidentyfikowane jako negatywne i w rzeczywistości takie są, to klasyfikację takich danych umieszcza się w tabeli prawdziwie negatywne (PN), a jeśli zostały źle zaklasyfikowane to lokuje się je w rubryce fałszywie negatywne (FN). Przewidywana dokładność jest miarą skuteczności związaną z uczeniem maszynowym, określana jako trafność ACC (ang. accuracy) (1.24).

$$ACC = \frac{(PP + PN)}{(PP + PN + FP + FN)} \quad (1.24)$$

Do porównania użyto także miar: czułość TPR (ang. true positive rate)(1.25), precyzję PPV (ang. positive predictive value) (1.26) i współczynnik $F1$ (1.27).

$$TPR = \frac{PP}{(PP + FN)} \quad (1.25)$$

$$PPV = \frac{PP}{(PP + FP)} \quad (1.26)$$

$$F1 = 2 \cdot \frac{(PPV \cdot TPR)}{(PPV + TPR)} \quad (1.27)$$

$$FPR = \frac{FP}{(FP + PN)} \quad (1.28)$$

Przy zrównoważonym zestawie danych jednym z rozwiązań może być zastosowanie współczynnika błędu, który wynosi $1 - ACC$ [16].

Powyższe parametry stosuje się w wypadku, gdy predykcja jest zero-jedynkowa. W sytuacji, gdy klasyfikator zwraca wartości prawdopodobieństwa z ciągłego przedziału $[0,1]$, można zastosować analizę za pomocą krzywej ROC. Punkty na tej krzywej odpowiadają różnym wartościom progu, powyżej którego odpowiedź klasyfikatora traktuje się jako pozytywną. Na osi x odkładane są wartości swoistości FPR (1.28), a na osi y wartości czułości TPR (1.25). Idealna krzywa ROC zawierałaby punkt $(0,1)$. Oznacza to, że wszystkie dane są prawidłowo zaklasyfikowane. Prosta $y = x$ odpowiada klasyfikatorowi losowemu. Pole powierzchni pod krzywą ROC to AUC . Jest to użyteczny wskaźnik, który pozostaje niezależny od wybranego kryterium oraz prawdopodobieństwa.

W uczeniu z nadzorem często stosuje się metodę walidacji krzyżowej, w której do trenowania i testowania wykorzystywany jest cały zbiór danych. Dane dzielone są na k podzbiorów. Następnie w każdej k -iteracji stosowane jest do uczenia $(k - 1)$ podzbiorów a 1 podzbiór służy do testowania. Wyniki ewaluacji pochodzące z każdej iteracji są uśredniane. Zaletą tej metody jest to, że można wykorzystać cały zbiór do uczenia i testowania, a wadą to, że sieci uczą się wielokrotnie na tych samych danych, co wydłuża cały proces. Wyniki uzyskane tą metodą są wiarygodne.

Ważne jest, aby porównać zbudowany klasyfikator z innymi prostymi klasyfikatorami wykonując predykcję niezależnie od danych na wejściu. Mogą to być:

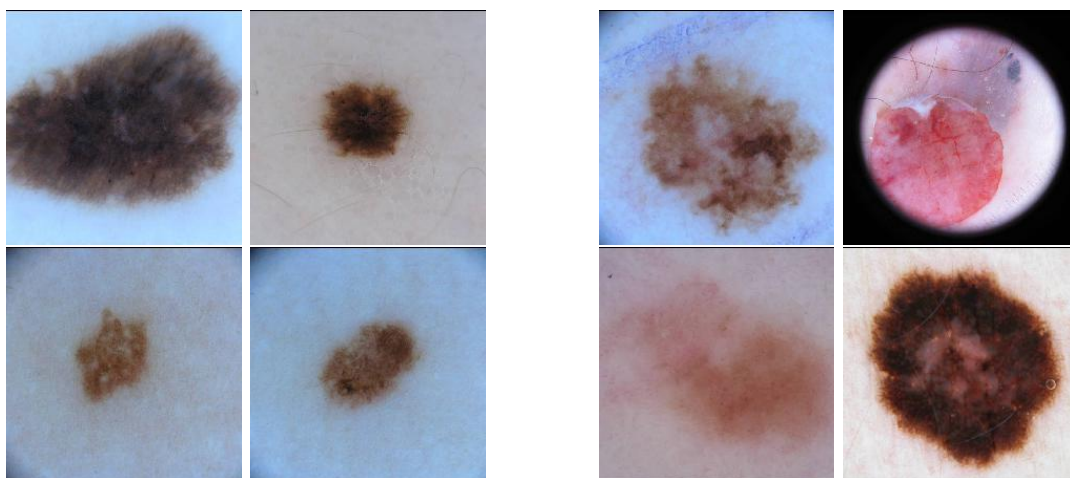
- klasyfikator zerowy - predykcja zawsze równa 0 ;
- klasyfikator losowy - predykcja zgodna z jednorodnym rozkładem prawdopodobieństwa;
- klasyfikator większościowy - predykcja zawsze równa klasie, której jest najwięcej w zbiorze danych.

Do porównywania najlepiej stosować kilka współczynników jednocześnie.

Rozdział 2

Dane eksperymentalne

W niniejszej pracy jako dane eksperymentalne użyto 900 zdjęć uzyskanych w dermatoskopie, a wykonanych aparatem fotograficznym zmian barwnikowych skóry w formacie JPEG. Obrazy te były podzielone na zmiany barwnikowe złośliwe (173 zdjęcia) oraz zmiany barwnikowe niezłośliwe (727 zdjęć). Obrazy zostały pobrane z International Skin Imaging Collaboration, ISIC [17]. Instytucja ta posiada jedną z największych na świecie baz danych zdjęć dermatologicznych. Baza danych składa się z obrazów, które sprawiają najwięcej trudności w klasyfikacji. Obrazy posiadały różny wymiar od 767 x 576 do 4288 x 2848 pikseli. Na rysunku przedstawiono przykładowe zmiany skórne z danej klasy, Rysunek 2.1.

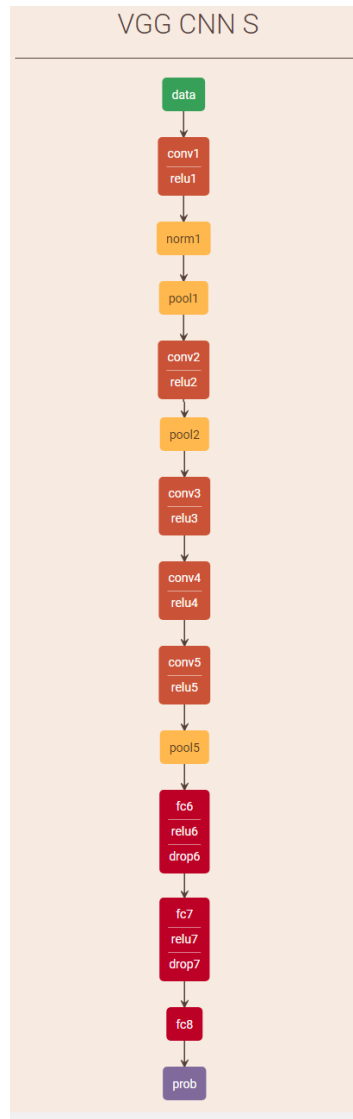


Rysunek 2.1: Przykłady obrazów zmian barwnikowych niezłośliwych po lewej i złośliwych po prawej.

Dane eksperymentalne zawierają czterokrotną przewagę zestawu danych składającą się ze zmian łagodnych. W tym przypadku testowanie danych daje niejednoznaczne wyniki. W tym celu odniesiono się do pracy [16], w której zauważono, że większość eksperymentów odnosi się do zestawu danych nie zrównoważonych, gdzie mniejsza ilość danych odnosi się do tych "ciekawszych" przypadków. W tym celu zrównoważono zestawy danych, aby zmniejszyć ilość danych "mniej ciekawych" w stosunku do ilości danych "ciekawych". Zastosowanie tej metody poprawia wyniki klasyfikatora.

Dodatkowo w pracy użyto wytrenowaną wcześniej sieć konwolucyjną udostępnioną przez BVLC (Berkeley Vision and Learning Center). Modele pochodziły z Image Large Scale Visual Recognition Challenge, ILSVRC. Sieci te zostały wytrenowane na bazie danych z ImageNet,

zawierające fotografie z 1000 różnych obiektów [18]. Model sieci wykorzystany w niniejszej pracy nazywa się VGG-CNN-S, i został wytworzony na potrzeby ILSVRSC 2012 [19]. Sieć zbudowana jest z 5 warstw konwolucyjnych - CONV i 3 warstw w pełni połączonych - FC. Dwie warstwy FC o długości 4096 i jednej o długości 1000, która stanowi wyjście z sieci. Model VGG-CNN-S był uczony na obrazach o rozmiarze 224x224 w formacie RGB, czyli 224x224x3, gdzie każdy z 3 kanałów odpowiada wartości 1 z 3 kanałów - czerwony, zielony, niebieski. Model przedstawiono na Rysunku 2.2.



Rysunek 2.2: Na rysunku przedstawiony jest model VGG-CNN-S zbudowany z 5 warstw konwolucyjnych CONV i 3 warstw w pełni połączonych FC.

Rozdział 3

Metodologia

3.1. Oprogramowanie

Programy, które użyto w ramach pracy napisano w języku Python 2.7. Biblioteki, które użyto to NumPy, Caffe, scikit-learn, Image. Dzięki zastosowaniu biblioteki Image można było dokonać transformacji na obrazach. Biblioteka Caffe dedykowana jest do uczenia głębokiego. Została ona stworzona i rozwinięta na BVLC. Dzięki tej bibliotece można konstruować i budować sieci neuronowe. Najczęściej stosuje się je do obróbki obrazów. Biblioteka ta zawiera interfejsy z użyciem języka Python. Istnieje wiele wytrenowanych modeli z użyciem tej biblioteki. Biblioteka scikit-learn zawiera gotowe funkcje wykorzystywane w uczeniu maszynowym z nadzorem i bez nadzoru. Są to zaimplementowane algorytmy, które zostały wykorzystane w niniejszej pracy. Scikit-learn korzysta z bibliotek NumPy oraz SciPy, dzięki czemu może wykonywać obliczenia macierzowe i statystyczne.

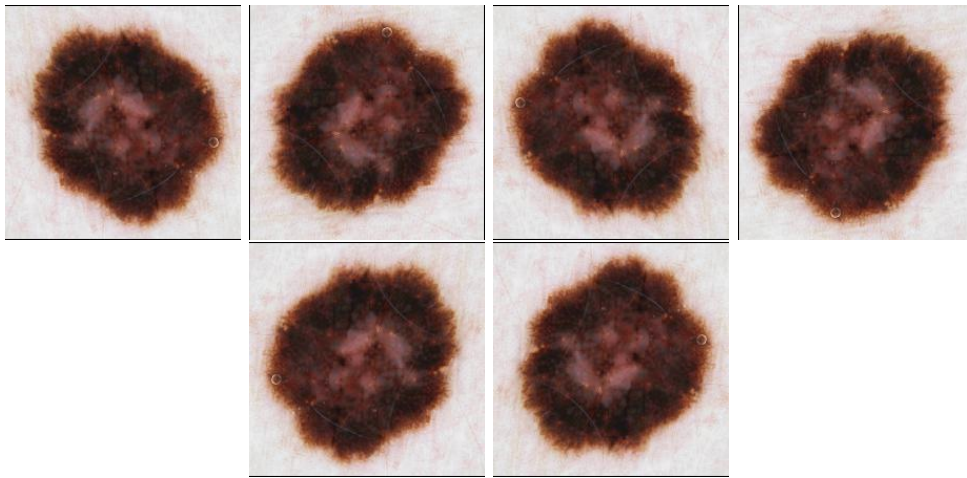
3.2. Klasyfikacja zdjęć. Metoda pracy

Wytworzono sztuczne obrazy uczące bazując na oryginalnym zestawie danych. Na wstępie obrazy przycięto do kwadratu w taki sposób, aby został zachowany środek obrazu (założono, że zmiana znajduje się w większości przypadków na środku obrazu). Następnie przeskalowano obraz do rozmiarów 224x224 tak, aby można było przepuścić obraz przez wytrenowaną sieć. Obraz musiał być przycięty wcześniej do kwadratu, gdyż samo przeskalowanie obrazu do odpowiedniego wymiaru skutkowało gorszymi wynikami klasyfikacji, niż w przypadku ewentualnego przycięcia zmiany skórnej na obrazie. Dodatkowo obrazy, które zostałyby tylko przeskalowane straciłyby swoje pierwotne proporcje, a kształt zmiany skórnej jest brany pod uwagę przez lekarza specjalistę przy identyfikacji zmiany jako złośliwej lub niezłośliwej. Ważne jest, aby cała zmiana skórna była widoczna na zdjęciu, ponieważ krawędzie zmiany także są przedmiotem badania dermoskopowego. Założono, że zmiany skórne mieszczą się na zdjęciu w większości przypadków. Na tak przygotowanych obrazach dokonano dalszych modyfikacji. Obrazy przycięte i przeskalowane będą dalej nazywane obrazami oryginalnymi lub bez obróbki.

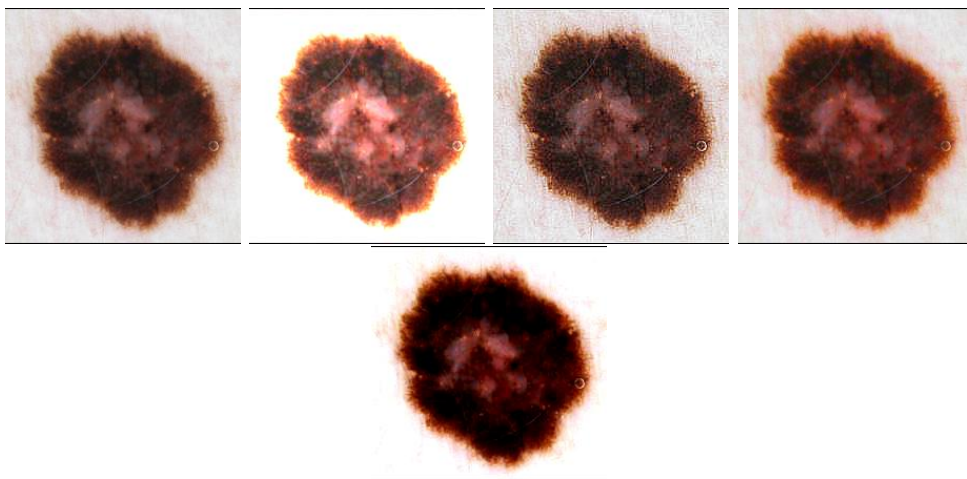
Do wytworzenia sztucznych obrazów uczących przeprowadzono transformacje, do których używane były tylko oryginalne obrazy. Po każdej transformacji powstawały nowe zbiory danych. W pierwszej kolejności zastosowano geometryczne transformacje obrazów. Dokonano poziomego odbicia obrazów, po czym te same obrazy oryginalne poddano odbiciom pionowym, do tego użyto funkcji Image.FLIP_LEFT_RIGHT oraz Image.FLIP_TOP_BOTTOM. Stosując funkcję Image.ROTATE_wartość_rotacji, Zastosowano także obrót obrazu o 90°,

180°, 270° Gdzie w miejsce wartość_rotacji wstawiono odpowiednią liczbę. Przykładowe przekształcenia geometryczne przedstawiono na Rysunku 3.1.

Następnie zastosowano przetworzenia kolorystyczne, które zwiększyły jasność obrazu przy pomocy funkcji Image.Enhance.Brightness z parametrem 2 (rozjaśniono obraz). Wytworzono nowe zbiory obrazów zwiększając intensywność kolorów obrazu funkcją Image.Enhance.Color o współczynniku 1.5, ostrość funkcją Image.Enhance.Sharpness o współczynniku 5 oraz kontrast funkcją Image.Enhance.Contrast ze współczynnikiem 1.5. Funkcja Image.Enhance o współczynniku 1 to oryginalny obraz, na którym nie ma żadnych zmian. Zwiększając ten współczynnik zwiększa się wartość wcześniej wymienionych parametrów, a zmniejszając go zmniejsza się ich wartość. Przedstawiono przykładowe przekształcenia oryginalnego obrazu 3.2.



Rysunek 3.1: Przekształcenia geometryczne od lewej obraz oryginalny, rotacja90, rotacja180, rotacja270, lewo_prawo, góra_dół.



Rysunek 3.2: Przekształcenia kolorów od lewej obraz oryginalny, zwiększona jasność, ostrość, intensywność koloru, kontrast.

Jak wspomniano wyżej model VGG-CNN-S przystosowany został do obrazów o wymiarach 224x224x3. Dlatego przed podaniem obrazu na wejście sieci przeskalowano obrazy do wymiaru 224x224 pikseli oraz odjęto odpowiednie wartości średnie od każdego z kanałów

RGB. Następnie przeprowadzono propagację przednią przez sieć, gdzie na wejście podany był obraz. Z sieci wyodrębniono warstwę szóstą w pełni połączoną o długości 4096, która dalej będzie zwana FC6. Jako wynik powyższej operacji uzyskano wektor o długości 4096. W ten sposób otrzymano sparametryzowany obraz w postaci wektora służący jako deskryptor obrazu.

W kolejnym etapie obraz w postaci wektora klasyfikowano z użyciem algorytmu SVM. Przed procesem uczenia wektor cech przeskalowano z użyciem funkcji logistycznej

$$x'_j = \frac{1}{1 + \exp^{-x_j}} \quad (\text{dla } j = 1, 2, \dots, n - 1, n) \quad (3.1)$$

gdzie x'_j - nowa przeskalowana wartość cechy, x_j - oryginalna wartość cechy, n jest równe 4096 (rozmiar warstwy FC6). Klasyfikator był budowany z użyciem odpowiedniego parametru C, który maksymalizował wartość współczynnika F1 oraz kernelem gausowskim (rbf). Ewaluację przeprowadzono stosując walidację krzyżową na 10 podzbiorach. W tym celu dzielono obrazy na dane treningowe i testowe. Dane testowe zawsze składały się tylko z obrazów oryginalnych.

Rozdział 4

Wyniki

W zbiorze danych znajduje się 900 obrazów. 173 obrazy zmian barwnikowych złośliwych przyjmuje wartość binarną 1 i 727 zmian barwnikowych niezłośliwych przyjmujących wartość binarną 0. Stosując walidację krzyżową obrazy dzielono na treningowe - 810 i testowe - 90. Z danych treningowych i testowych tworzone zbiory zrównoważone i niezrównoważone. Zbiory zrównoważone posiadają tę samą liczbę obrazów z etykietą 0 i 1. Zbiory niezrównoważone mają różne ilości 0 i 1. Rozdział został podzielony na dwa podrozdziały. Pierwszy odpowiada klasyfikatorowi, w którym dane treningowe posiadają niezrównoważone zbiory danych. Drugi podrozdział odpowiada klasyfikatorowi, w którym dane treningowe posiadają zrównoważone zbiory. Testy były wykonane na zbiorach zrównoważonych i niezrównoważonych dla każdego podrozdziału.

4.1. Dane treningowe budowane na zbiorze niezrównoważonym

Rozpoczęto pracę od wykonania klasyfikatora losowego i zerowego na danych treningowych budowanych na zbiorze niezrównoważonym. Wyniki przedstawiono w Tabeli 4.1.

| | klasyfikator zerowy | klasyfikator losowy |
|----------|---------------------|---------------------|
| Trafność | 0.807 ± 0.074 | 0.499 ± 0.033 |
| F1 | 0 | 0.258 ± 0.092 |
| Czułość | 0 | 0.48 ± 0.12 |
| Precyzja | 0 | 0.186 ± 0.083 |

Tabela 4.1: Wyniki dla klasyfikatora zerowego i losowego dla danych treningowych niezrównoważonych.

Następnie wykonano klasyfikator SVM na danych oryginalnych testowanych na dwóch testach. Pierwszy test budowany na zbiorze niezrównoważonym, drugi na zbiorze zrównoważonym. Wyniki w Tabeli 4.2. Dane treningowe zawierają wszystkie dostępne obrazy w ilości 810, a dane testowe dla testu niezrównoważonego 90. W tych zbiorach zawarte są losowe liczby wystąpień 0 i 1. W danych testowych dla zbiorów zrównoważonych najpierw algorytm zliczał ilość 1, znajdował obrazy z tą etykietą. Następnie znajdował obrazy z etykietą 0 i dobierał taką samą ich ilość, co obrazów z etykietą 1. Parametr C był dobierany w ten sposób by maksymalizować współczynnik F1.

Na podstawie tabeli 4.2 można zauważyć, że współczynnik F1 i precyzja jest wyższy dla testu budowanego na zbiorze zrównoważonym. Czułość prezentuje się na tym samym poziomie.

| klasyfikator SVM | dane testowe niezrównoważone | dane testowe zrównoważone |
|------------------|------------------------------|---------------------------|
| Trafność | 0.814 ± 0.067 | 0.665 ± 0.076 |
| F1 | 0.43 ± 0.12 | 0.52 ± 0.15 |
| Czułość | 0.39 ± 0.15 | 0.39 ± 0.15 |
| Precyzja | 0.56 ± 0.22 | 0.88 ± 0.12 |

Tabela 4.2: Wyniki dla klasyfikatora SVM kernel rbf C=130,130 dla danych treningowych, oryginalnych, budowanych na zbiorze niezrównoważonym.

Wcześniej wspomniano, że należy najpierw przyciąć obraz do kwadratu, a następnie przeskalować obraz. Przeprowadzono eksperyment mający na celu sprawdzenie, co stanie się, jeśli obrazy zostaną przeskalowane bez przycięcia. Wyniki w Tabeli 4.3. Tak jak założono, samo skalowanie obrazu zniekształca go. Porównując tabelę 4.2 kolumnę pierwszą z tabelą 4.3 widać, że wyniki w wypadku samego przeskalowania obrazów są gorsze. Trafność i czułość są niższe, niż w wypadku klasyfikatora zerowego i losowego, co uzasadnia stosowanie tylko danych uprzednio przyciętych do kwadratu.

| klasyfikator SVM | |
|------------------|-------------------|
| Trafność | 0.788 ± 0.071 |
| F1 | 0.360 ± 0.091 |
| Czułość | 0.315 ± 0.097 |
| Precyzja | 0.48 ± 0.22 |

Tabela 4.3: Wyniki dla klasyfikatora SVM kernel rbf C=180 dane treningowe i testowe niezrównoważone.

Następnie sprawdzono, jak prezentują się zmiany wykonane na obrazach oryginalnych. W tym celu dane treningowe zostały zwielokrotnione transformacjami. Zbiór danych testowych składa się tylko z obrazów oryginalnych. Ich ilość wynosi 90 zarówno w zbiorach niezrównoważonych jak i w zbiorach zrównoważonych $2 * ilosc_wystapien_1$. W pierwszym wypadku zbiór danych treningowych zwielokrotniano czterokrotnie ($4 * 810$). Dane treningowe zawierają obrazy oryginalne, obrazy obrócone o 90° , obrazy obrócone o 180° , obrazy obrócone o 270° . Drugi sposób to zwielokrotnianie obrazów pięciokrotnie ($5 * 810$). W danych testowych znajdować się będą obrazy oryginalne, obrazy ze zwiększoną intensywnością koloru, ostrości, kontrastu i jasności. Wyniki zostały przedstawione w Tabeli 4.4. Tak jak w przypadku

| klasyfikator SVM | dane testowe niezrównoważone | | dane testowe zrównoważone | |
|------------------|--|---|--|---|
| | rotacja 90 rotacja180 rotacja270 | kolor ostrość kontrast jasność | rotacja 90 rotacja180 rotacja270 | kolor ostrość kontrast jasność |
| Trafność | 0.808 ± 0.083 | 0.811 ± 0.067 | 0.648 ± 0.090 | 0.651 ± 0.059 |
| F1 | 0.41 ± 0.15 | 0.42 ± 0.13 | 0.51 ± 0.15 | 0.50 ± 0.14 |
| Czułość | 0.36 ± 0.15 | 0.38 ± 0.16 | 0.38 ± 0.15 | 0.37 ± 0.14 |
| Precyzja | 0.57 ± 0.25 | 0.52 ± 0.19 | 0.84 ± 0.20 | 0.86 ± 0.11 |

Tabela 4.4: Wyniki dla klasyfikatora SVM kernel rbf i odpowiednio C=50,60,140,90 dla danych treningowych niezrównoważonych. Zbiory uczące były powiększone o zbiory z różnymi transformacjami.

zbiorów oryginalnych dane testowe zawierające zbiory zrównoważone wydają się być lepsze. Współczynnik F1 i wartość precyzji są większe dla tego zbioru. Po dodaniu sztucznych obrazów uczących zauważono, że wartości współczynników są do siebie bardzo podobne. Na podstawie tych wyników nie jesteśmy w stanie stwierdzić, że zmiany które wprowadziliśmy w obrazach były skuteczną metodą poprawy klasyfikatora.

W dalszej kolejności sprawdzono, jaki wpływ mają pojedyncze zmiany na zbiór oryginalny w przypadku danych treningowych i testowych niezrównoważonych. Pierwotnie pracowano tylko na danych niezrównoważonych. Porównano wyniki z danymi oryginalnymi w tabeli 4.2. Sztucznie powiększono zestaw danych uczących o rotację obrazów o 90°, 180°, 270°, odbicie względem osi pionowej(lewo-prawo), odbicie względem osi poziomej(góra-dół). W zestawie danych uczących występuje teraz podwójna liczba danych(1620): danych oryginalnych i danych poddanych transformacji. W danych testowych znajdują się tylko dane oryginalne (90), Tabela 4.5.

| klasyfikator SVM | Rotacja 90° | rotacja 180° | rotacja 270° | lewo,prawo | góra,dół |
|------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| Trafność | 0.833 ± 0.057 | 0.816 ± 0.075 | 0.813 ± 0.079 | 0.811 ± 0.066 | 0.803 ± 0.073 |
| F1 | 0.39 ± 0.15 | 0.41 ± 0.15 | 0.43 ± 0.15 | 0.40 ± 0.11 | 0.36 ± 0.17 |
| Czułość | 0.30 ± 0.14 | 0.34 ± 0.15 | 0.38 ± 0.15 | 0.34 ± 0.12 | 0.31 ± 0.19 |
| Precyzja | 0.62 ± 0.26 | 0.58 ± 0.26 | 0.57 ± 0.25 | 0.55 ± 0.24 | 0.48 ± 0.25 |

Tabela 4.5: Wyniki dla klasyfikatora SVM kernel rbf i odpowiednio C= 40,60,60,70,90 dla danych niezrównoważonych. Zbiór uczący powiększony o sztuczne obrazy z zastosowaniem transformacji geometrycznych.

Tak samo powiększono zestaw danych o zmianę intensywności koloru, kontrastu, ostrości i jasności, zwiększając zbiór uczący, tabela 4.6.

| klasyfikator SVM | kolor | ostrość | kontrast | jasność |
|------------------|-------------------|-------------------|-------------------|-------------------|
| Trafność | 0.813 ± 0.062 | 0.811 ± 0.074 | 0.813 ± 0.072 | 0.811 ± 0.063 |
| F1 | 0.40 ± 0.15 | 0.44 ± 0.15 | 0.42 ± 0.15 | 0.43 ± 0.020 |
| Czułość | 0.35 ± 0.16 | 0.41 ± 0.18 | 0.36 ± 0.17 | 0.39 ± 0.090 |
| Precyzja | 0.55 ± 0.27 | 0.53 ± 0.18 | 0.58 ± 0.27 | 0.55 ± 0.18 |

Tabela 4.6: Wyniki dla klasyfikatora SVM kernel rbf i odpowiednio C= 50,90,70,100 dla danych niezrównoważonych. Zbiór uczący powiększony o sztuczne obrazy uczące po transformacji kolorystycznej.

W pierwszej kolejności zauważono, że wyniki te są zbliżone do wartości klasyfikatora oryginalnego. Zmiany te nie wniosły żadnych istotnych dodatkowych cech do klasyfikatora. Zauważono także, że w wyniku rotacji obrazu o 90° i rotacji o 270° wyniki są różne od siebie pomimo, że są to lustrzane odbicia. Nie znaleziono wytłumaczenia tego zjawiska. Podczas kilkukrotnego sprawdzania danych uzyskano różne wyniki, które w granicach błędu statystycznego są podobne.

4.2. Dane treningowe budowane na zbiorze zrównoważonym

Tak jak wspomniano wcześniej dane treningowe były budowane za zbiorze zrównoważonym. To znaczy, że dane treningowe miały tę samą ilość etykiet 1 i 0. Testy były wykonane na zbiorach niezrównoważonym i zrównoważonym. Na początek zbudowano klasyfikator zerowy

| | klasyfikator zerowy | klasyfikator losowy |
|----------|---------------------|---------------------|
| Trafność | 0.5 | 0.525 ± 0.078 |
| F1 | 0 | 0.516 ± 0.070 |
| Czułość | 0 | 0.505 ± 0.069 |
| Precyzja | 0 | 0.533 ± 0.088 |

Tabela 4.7: Wyniki dla klasyfikatora zerowego i losowego dla danych zrównoważonych.

i losowy na zbiorze zrównoważonym. Wyniki w Tabeli 4.7. Następnie wykonano klasyfikację na danych oryginalnych tabela 4.8.

| klasyfikator SVM | dane testowe niezrównoważone | dane testowe zrównoważone |
|------------------|------------------------------|---------------------------|
| Trafność | 0.71 ± 0.13 | 0.67 ± 0.10 |
| F1 | 0.49 ± 0.12 | 0.65 ± 0.13 |
| Czułość | 0.70 ± 0.13 | 0.64 ± 0.19 |
| Precyzja | 0.41 ± 0.19 | 0.71 ± 0.14 |

Tabela 4.8: Wyniki dla klasyfikatora SVM na danych oryginalnych, kernel rbf C=10,30 dla danych treningowych zrównoważonych.

Porównując ze sobą współczynnik F1 oraz precyzję można stwierdzić, że klasyfikator, który uczy się i jest testowany na zbiorze zrównoważonym, ma najlepsze wyniki.

Tak samo jak w poprzednim podrozdziale zastosowano zwielokrotnione zastawy danych. W tym wypadku najpierw odnaleziono obrazy z etykietą 1. Następnie zwiększono ilość danych o odpowiednie transformacje: zwiększenie intensywności koloru, ostrości, kontrastu, jasności oraz w drugim zbiorze o rotacje obrazu o 90, 180, 270 °. Następnie uzupełniono zbiór obrazami oryginalnymi z etykietą 0. Użyto całego zbioru obrazów oryginalnych z etykietą 0 oraz obrazów z etykietą 1 i dodatkowo obrazami sztucznymi występującymi w zestawie uczącym. Testy były wykonane na obrazach oryginalnych. Wyniki w Tabeli 4.9

| klasyfikator SVM | dane testowe niezrównoważone | | dane testowe zrównoważone | |
|------------------|--|---|--|---|
| | rotacja 90 rotacja180 rotacja270 | kolor ostrość kontrast jasność | rotacja 90 rotacja180 rotacja270 | kolor ostrość kontrast jasność |
| Trafność | 0.72 ± 0.11 | 0.809 ± 0.068 | 0.712 ± 0.099 | 0.679 ± 0.091 |
| F1 | 0.48 ± 0.12 | 0.46 ± 0.13 | 0.70 ± 0.11 | 0.653 ± 0.092 |
| Czułość | 0.67 ± 0.14 | 0.44 ± 0.14 | 0.67 ± 0.15 | 0.60 ± 0.11 |
| Precyzja | 0.41 ± 0.17 | 0.53 ± 0.21 | 0.77 ± 0.16 | 0.73 ± 0.14 |

Tabela 4.9: Wyniki dla klasyfikatora SVM kernel rbf i odpowiednio C=7,30,2,1 dla danych treningowych zrównoważonych. Zbiór uczący powiększony o obrazami o różnych transformacjach.

Zauważono, że przy teście niezrównoważonym, wyniki są gorsze, niż wyniki dla testu na oryginalnych danych. Oznacza to, że zmiany wykonane na obrazach nie są dobrą metodą tworzenia sztucznych danych uczących. Za to w wyniku użycia zbioru treningowego zrównoważonego oraz zbioru testowego zrównoważonego zauważono widoczną poprawę skuteczności klasyfikatora w przypadku transformacji z rotacją obrazów. Dla zbioru, gdzie nastąpiła trans-

formacja kolorystyczna, wyniki są podobne do wyników testów przeprowadzonych na danych oryginalnych. W dalszej kolejności wykonywano testy tylko na zbiorze zrównoważonym.

Następnie zbadano, jaki wpływ na skuteczność klasyfikatora mają pojedyncze transformacje przeprowadzone na obrazach. W tym celu w zbiorze uczącym najpierw zostały odnalezione obrazy z etykietą 1. Następnie przeprowadzono na tych obrazach transformacje i dodano je do zbioru uczącego. Zbiór był uzupełniony obrazami z etykietą 0 w ilości równej liczbie obrazów z etykietą 1 - obraz oryginalny i po transformacji. Po każdej transformacji budowano nowy zbiór zawierający obrazy oryginalne i po jednej, konkretnej transformacji, tak więc zbiorów było tyle samo, co rodzajów przeprowadzonych transformacji. Transformacje geometryczne przedstawiono w tabeli 4.11 a kolorystyczne w tabeli 4.10.

| klasyfikator SVM | kolor | ostrość | kontrast | jasność |
|------------------|-------------------|-------------------|-------------------|-------------------|
| Trafność | 0.726 ± 0.083 | 0.67 ± 0.10 | 0.711 ± 0.096 | 0.716 ± 0.095 |
| F1 | 0.738 ± 0.068 | 0.670 ± 0.094 | 0.722 ± 0.088 | 0.702 ± 0.090 |
| Czułość | 0.76 ± 0.13 | 0.68 ± 0.16 | 0.75 ± 0.11 | 0.67 ± 0.14 |
| Precyzja | 0.74 ± 0.14 | 0.71 ± 0.16 | 0.72 ± 0.13 | 0.77 ± 0.14 |

Tabela 4.10: Wyniki dla klasyfikatora SVM kernel rbf i odpowiednio C=5,16,4,14 dla danych zrównoważonych. Dane uczące zawierały sztuczne obrazy powiększone o pojedyncze transformacje kolorystyczne.

| klasyfikator SVM | Rotacja 90° | rotacja 180° | rotacja 270° | lewo,prawy | górną,dół |
|------------------|-------------------|-----------------|--------------------|-----------------|-------------------|
| Trafność | 0.70 ± 0.10 | 0.69 ± 0.13 | 0.698 ± 0.0097 | 0.70 ± 0.12 | 0.70 ± 0.11 |
| F1 | 0.728 ± 0.092 | 0.73 ± 0.12 | 0.722 ± 0.095 | 0.73 ± 0.10 | 0.727 ± 0.089 |
| Czułość | 0.79 ± 0.12 | 0.80 ± 0.14 | 0.79 ± 0.14 | 0.80 ± 0.11 | 0.79 ± 0.13 |
| Precyzja | 0.69 ± 0.14 | 0.69 ± 0.17 | 0.69 ± 0.12 | 0.68 ± 0.14 | 0.70 ± 0.14 |

Tabela 4.11: Wyniki dla klasyfikatora SVM kernel rbf i odpowiednio C= 3,8,6,5,5 dla danych zrównoważonych. Dane uczące zawierały sztuczne obrazy powiększone o pojedyncze transformacje geometryczne.

Zauważono, że każda z pojedynczych transformacji przyczyniła się do zwiększenia skuteczności klasyfikatora. Wszystkie współczynniki są większe niż na oryginalnym zbiorze danych. Najlepszą transformacją kolorystyczną okazała się zmiana intensywności koloru, a najmniej korzystną zmiana ostrości obrazu. Transformacje geometryczne prezentują się na podobnym, wysokim poziomie.

Po przeanalizowaniu tych Tabel stwierdzono, że jest możliwe zwiększenie skuteczności klasyfikatora w zbiorze z kilkukrotnym zwiększeniem zbioru danych uczących poprzez usunięcie transformacji z ostrością obrazu, a pozostawieniem zmian intensywności w kolorze, kontraście i jasności. Kolejnym pomysłem było zastosowanie na jednym obrazku rotacji oraz zmian intensywności w kolorze. Wyniki testów dla tego klasyfikatora przedstawia Tabela 4.12 w przypadku transformacji z kolorami i Tabeli 4.13 w przypadku rotacji i zmian intensywności koloru na jednym obrazku.

Zauważono, że w przypadku odjęcia zmian w ostrości obrazu, klasyfikator zwiększył skuteczność, co pokazuje poprawę wszystkich ocen klasyfikatora. W przypadku dwóch zmian na jednym obrazku okazało się, że klasyfikator jest mniej wydajny niż w przypadku samej rotacji obrazu 4.9.

| klasyfikator SVM | dane testowe zrównoważone |
|------------------|---------------------------|
| Trafność | 0.70 ± 0.11 |
| F1 | 0.68 ± 0.12 |
| Czułość | 0.67 ± 0.16 |
| Precyzja | 0.72 ± 0.13 |

Tabela 4.12: Wyniki dla klasyfikatora SVM kernel rbf C=0.2 dla danych treningowych zrównoważonych. Zbiór uczący zawierał dane oryginalne, sztuczne obrazy po transformacji w intensywności koloru, jasności i kontrastu.

| klasyfikator SVM | dane testowe zrównoważone |
|------------------|---------------------------|
| Trafność | 0.67 ± 0.13 |
| F1 | 0.68 ± 0.13 |
| Czułość | 0.71 ± 0.18 |
| Precyzja | 0.67 ± 0.13 |

Tabela 4.13: Wyniki dla klasyfikatora SVM kernel rbf C=0.2 dla danych treningowych zrównoważonych. Zbiór uczący składał się z obrazów oryginalnych i sztucznych obrazów gdzie na jednym obrazie znajdowała się rotacja odpowiednio o 90, 180 i 270 ° wraz ze zmianą w intensywności koloru.

4.3. Krzywa ROC

Pierwszym warunkiem, aby wytworzyć krzywą ROC jest uzyskanie odpowiedzi systemu w formie wartości ze zbioru ciągłego z przedziału (0,1). Biblioteka sklearn.svm standardowo zwraca wartości dyskretne 0 lub 1. Aby wytworzyć krzywą ROC posłużono się funkcją z biblioteki sklearn.metrics.roc_curve(), która w swoich argumentach przyjmuje przewidziane wartości w postaci prawdopodobieństwa wystąpienia danej klasy oraz prawidłową przynależną klasę. Wartości prawdopodobieństwa zostały obliczone za pomocą funkcji sklearn.svm.libsvm.predict_proba(). Funkcja zwraca wartości z przedziału (0,1).

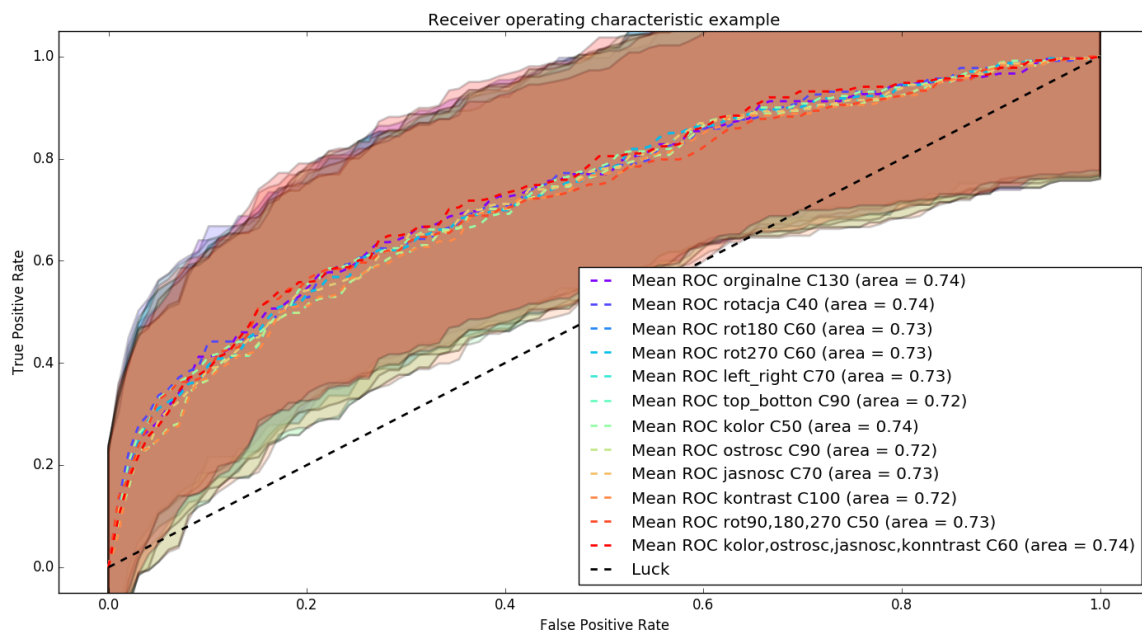
Krzywe ROC dla zbiorów uczących i testowych niezrównoważonych została przedstawiona na Rysunku 4.1.

Krzywe ROC dla zbioru uczącego i testowego zrównoważonego na Rysunku 4.2. Zauważono, że nie ma różnic między krzywymi ROC dla danych uczących powiększonych o różne transformacje. Można przypuszczać, że dla zbiorów zrównoważonych lepsze byłoby zastosowanie zbioru danych z większą ilością zmian w zbiorze uczącym. Pod krzywą dla zbioru z największą ilością danych pole pod powierzchnią jest największe (0.80, 0.81). Trudno jest zinterpretować wyniki przedstawione na krzywych ROC. Wartości prawdopodobieństwa zostały sztucznie wytworzone za pomocą funkcji sklearn.svm.libsvm.predict_proba().

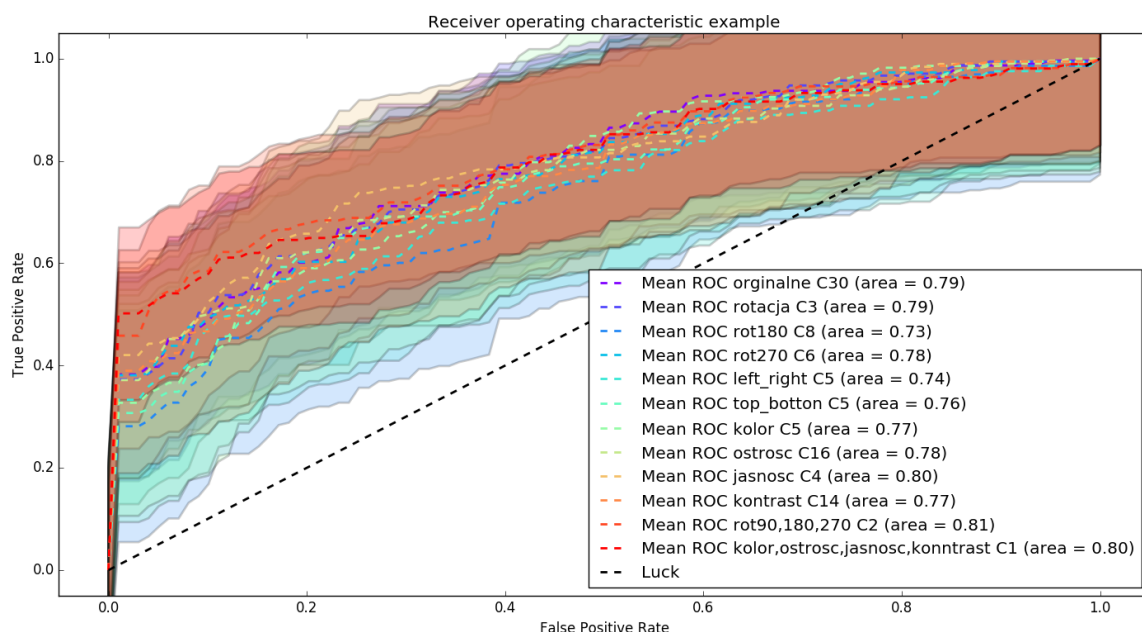
4.4. Wnioski

Niezależnie od sposobu testowania, użycie zrównoważonego zbioru uczącego daje lepsze wyniki. Jest to zgodne z pracą [16]. Sztuczne powiększanie zbioru danych to dobry sposób na budowanie zbioru uczącego.

Zwiększając zbiór uczący przy pomocy kilku transformacji należy wcześniej sprawdzić, jak każda z nich wpływa na klasyfikator i wybrać te, które mogą zwiększyć jego skuteczność.. Tworząc sztuczne obrazy uczące zauważono, że skuteczniejsze jest stosowanie transformacji



Rysunek 4.1: Krzywa ROC dla danych niezrównoważonych. W opisie legendy przedstawione są dodatkowe, sztuczne obrazy poza oryginalnymi, które wystąpiły w zbiorze uczącym, przy odpowiednim parametrze C oraz zostało wyliczone AUC.



Rysunek 4.2: Krzywa ROC dla danych zrównoważonych. W opisie legendy przedstawione są dodatkowe, sztuczne obrazy poza oryginalnymi, które wystąpiły w zbiorze uczącym, przy odpowiednim parametrze C oraz zostało wyliczone AUC.

geometrycznych. W przypadku transformacji kolorystycznych skuteczne okazały się transformacje polegające na zmianie intensywności koloru, kontrastu i jasności. Zmiana ostrości obrazu nie poprawiła skuteczności klasyfikatora. Współczynniki zastosowane do transformacji kolorystycznych były losowe. Prawdopodobnie zastosowanie współczynnika intensywności ostrości o innej wartości niż użyta w niniejszej pracy mogłoby zmienić wyniki uzyskane przez klasyfikator.

W wypadku uczenia na zbiorze niezrównoważonym optymalny współczynnik C jest rzędu 100, a dla zbioru zrównoważonego jego wartość jest dużo mniejsza i spada nawet do 0.2. W zrównoważonym zbiorze uczącym, zbudowanym w sposób opisany w niniejszej pracy (mała liczba danych, mniej zróżnicowany od niezrównoważonego), występuje konieczność zwiększenia znaczenia regularyzacji, aby uniknąć przetrenowania.

Kiedy dane uczące są zrównoważone, a dane testowe niezrównoważone, wtedy klasyfikator SVM jest bardzo niestabilny. Metody, dzięki którym oceniano klasyfikator, przy każdym przepuszczeniu obrazów przez niego wskazywały inne wartości współczynników $F1$, czułości, precyzji i dokładności. Takie połączenie danych uczących i testowych nie daje wiarygodnych wyników.

Porównując wyniki klasyfikatora, wyższe wyniki uzyskano dla zastosowania do uczenia zbiorów powiększonych o pojedyncze zmiany wykonane na obrazach niż zbiorów powiększonych o kilka transformacji równocześnie.

Zwrócono uwagę na fakt, że błąd statystyczny zawierał się w przedziale 0.09 – 0.20. Zaobserwowane różnice między klasyfikatorami znalazły się w granicach błędu. Jakkolwiek udało się zauważyć pewne tendencje, nie można stwierdzić, że zostały one statystycznie udowodnione. Na podstawie wstępnego rozeznania wydaje się, że można byłoby zmniejszyć błąd poprzez:

- zmniejszenie liczby podzbiorów w walidacji krzyżowej
- ustalenie stałej wielkości zrównoważonego zbioru testowego
- zmniejszenie ilości cech wybierając te z nich, które najbardziej różnicują skórne zmiany barwnikowe złośliwe od niezłośliwych w oparciu o średnie wartości parametrów w tych grupach lub stosując test Kołmogorowa-Smirnowa. W Pythonie znajduje się funkcja z biblioteki `scipy.stats.ks_2samp`. Przyjmuje ona dane z dwóch grup testowych i zwraca wektor cech istotnych na ustalonym poziomie istotności. Jej zastosowanie pomogłoby wybrać cechy najistotniejsze dla różnicowania zmian złośliwych i niezłośliwych.

W wielu pracach niepewność statystyczna nie jest liczona, jednak jej podanie jest konieczne do stwierdzenia, czy wyniki są istotne statystycznie. Odchylenie standardowe uzyskane w niniejszej pracy jest zbyt duże i opisane powyżej modyfikacje mogłyby prowadzić do jego zmniejszenia.

Rozdział 5

Podsumowanie

W pracy skupiono się na sprawdzeniu możliwości transformacji obrazów w celu wytworzenia sztucznych obrazów uczących oraz na sposobie uczenia i testowania danych. Mając na celu wytrenowanie klasyfikatora skutecznego w rozpoznawaniu czerniaka, należy przygotować jak najlepszy zbiór danych uczących. Aby to uzyskać należy uwzględnić algorytmy stosowane przez lekarzy dermatologów. Ponadto zastosowanie prostych transformacji geometrycznych oraz transformacji kolorystycznych może okazać się pomocne przy tworzeniu zbioru obrazów uczących. Użycie zrównoważonych zbiorów uczących zwiększa skuteczność klasyfikatora. Wyniki testów są bardziej jednoznaczne i zrozumiałe przy zastosowaniu zrównoważonych zbiorów testowych. Jednak w praktyce dermatolog częściej ogląda zmiany barwnikowe niezłośliwe niż złośliwe, gdyż występują one u ludzi dużo częściej. Dlatego testowanie na zbiorach niezrównoważonych również powinno być wzięte pod uwagę. Wytwarzanie sztucznych obrazów uczących może być dobrą metodą zwiększenia skuteczności klasyfikatora w przypadku niewystarczającej liczby danych, jednakże w pierwszej kolejności należy korzystać z jak największej ilości oryginalnych danych.

Bibliografia

- [1] Cristina Nader Vasconcelos and Bárbara Nader Vasconcelos. Increasing deep learning melanoma classification by classical and expert knowledge based image transforms. *CoRR*, abs/1702.07025, 2017.
- [2] Sławomir Majewski Stefania Jabłońska. *Choroby skóry i choroby przenoszone drogą płciową*, pages 411–416. Wydawnictwo Lekarskie PZWL, Warszawa, 2010.
- [3] Ammara Masood and Adel Ali Al-Jumaily. Computer aided diagnostic support system for skin cancer: A review of techniques and algorithms. *International Journal of Biomedical Imaging*, 2013, Article ID 323268, 22 pages, 2013. doi:10.1155/2013/323268, 2013.
- [4] Luc Thomas and Ralph P.Braun. *Algorytmy diagnostyczne zmian melanocytowych*, pages 29–58. 2008.
- [5] Piotr Urbanek. *Sztuczne sieci neuronowe*, 2007.
- [6] Justin Johnson Andrej Karpathy. Cs231n: Convolutional neural networks for visual recognition notes. Stanford University, U.S.A.
- [7] Jason Weston. Support vector machine. In: Tutorial. 4 Independence Way, Princeton, USA.
- [8] George Kimeldorf and Grace Wahba. Some results on tchebycheffian spline functions. 33(1):82–95, 1971.
- [9] Sebastien C. Wong, Adam Gatt, Victor Stamatescu, and Mark D. McDonnell. Understanding data augmentation for classification: when to warp? *CoRR*, abs/1609.08764, 2016.
- [10] Cristina Nader Vasconcelos and Bárbara Nader Vasconcelos. Increasing deep learning melanoma classification by classical and expert knowledge based image transforms. *CoRR*, abs/1702.07025, 2017.
- [11] Deep image: Scaling up image recognition. *CoRR*, abs/1501.02876, 2015. Withdrawn.
- [12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In P. Bartlett, F.c.n. Pereira, C.j.c. Burges, L. Bottou, and K.q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1106–1114. 2012.
- [13] Francesco Visin, Kyle Kastner, Kyunghyun Cho, Matteo Matteucci, Aaron C. Courville, and Yoshua Bengio. Renet: A recurrent neural network based alternative to convolutional networks. *CoRR*, abs/1505.00393, 2015.

- [14] Fu Jie Huang and Yann LeCun. *Large-scale learning with SVM and convolutional nets for generic object categorization*, volume 1, pages 284–291. 2006.
- [15] Yichuan Tang. Deep learning using support vector machines. *CoRR*, abs/1306.0239, 2013.
- [16] Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.
- [17] Witryna internetowa international skin imaging collaboration: <http://isdis.net/isic-project/>.
- [18] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.
- [19] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the devil in the details: Delving deep into convolutional nets. In *British Machine Vision Conference*, 2014.