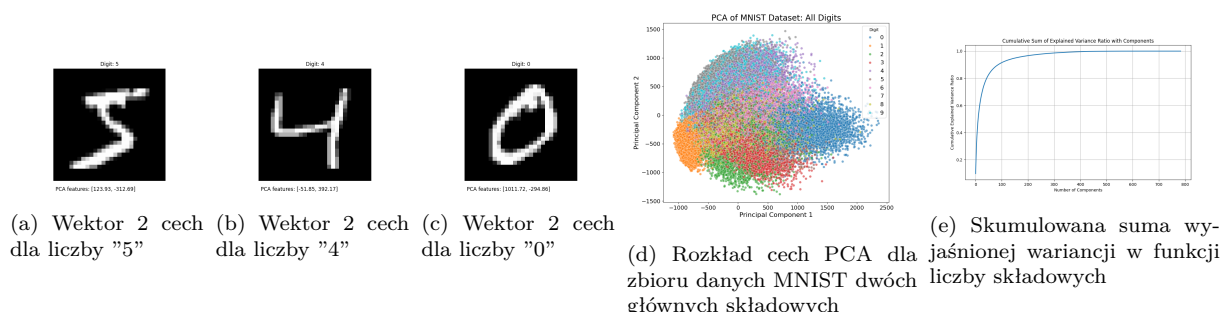


Obliczenia Inteligentne	Projekt 2 — Zadanie 1	
Grupa: Grupa 1	Dzień i czas: Czwartek, 10:00	Rok akademicki: 2023/24
Imię i nazwisko: JAKUB PAWLAK	Imię i nazwisko: MAGDALENA PAKUŁA	

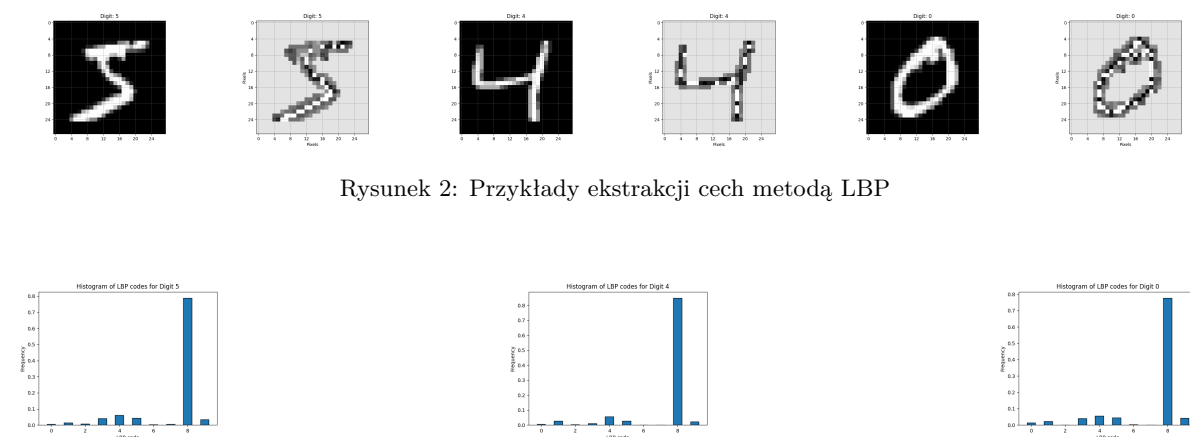
1 Opis ekstrakcji cech — Osoba 1 (Magdalena Pakuła)

Principal Component Analysis (PCA) - Analiza głównych składowych technika redukcji wymiarowości, która pomaga w przekształcaniu danych wielowymiarowych w postać niskowymiarową, przy jednoczesnym zachowaniu jak największej oryginalnej zmienności. Metoda PCA standaryzuje dane, oblicza macierz kowariancji rejestrującą relacje między cechami, a następnie dekomponuje ją na wartości własne i wektory własne. Wektory własne reprezentują kierunki największej wariancji, a wartości własne jej wielkość. Po posortowaniu wartości własnych malejąco, wybierane są największe z nich i odpowiadające im wektory własne do utworzenia nowej przestrzeni cech. Dane są rzutowane na tę przestrzeń, co skutkuje zbiorem danych o zredukowanej liczbie wymiarów, zachowującym najważniejsze informacje. W kontekście danych MNIST, PCA może być stosowana do redukcji liczby wymiarów obrazów cyfr, co ułatwia ich analizę i wizualizację, a także przyspiesza trening modeli uczenia maszynowego poprzez zmniejszenie ilości przetwarzanych danych przy jednoczesnym zachowaniu istotnych informacji.



Rysunek 1: Przykłady ekstrakcji cech metodą PCA

Binary Patterns (LBP) - Lokalne wzorce binarne to technika ekstrakcji cech używana głównie w analizie obrazów, która przekształca obraz w zestaw wartości binarnych opisujących teksturę. Proces ten polega na porównywaniu każdego piksela z jego sąsiadami. Piksel centralny jest traktowany jako próg, a każdy sąsiedni piksel jest porównywany do tego progu. Jeśli wartość sąsiada jest większa lub równa wartości centralnego piksela, przypisywana jest wartość 1, w przeciwnym razie 0. Wynikowe wartości binarne są następnie łączone w jednolitą wartość, która reprezentuje wzorec tekstur w danym obszarze obrazu. Metoda LBP jest odporna na zmiany jasności i kontrastu, co czyni ją użyteczną w różnych zastosowaniach, takich jak rozpoznawanie twarzy czy klasyfikacja tekstur. W przypadku danych MNIST, LBP może być używane do ekstrakcji cech, które są następnie wykorzystywane do klasyfikacji obrazów cyfr.



Rysunek 3: Histogramy LBP dla cyfry 0

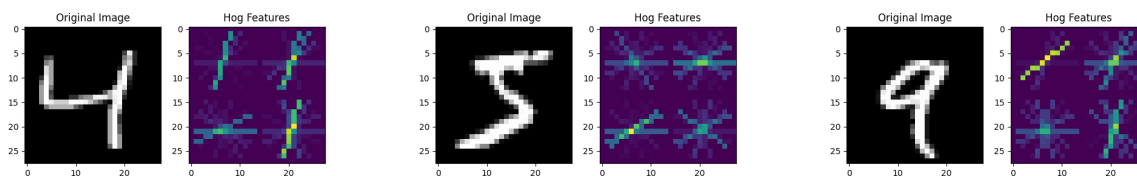
2 Wyniki eksperymentu — Osoba 1 (Magdalena Pakuła)

3 Opis ekstrakcji cech — Osoba 2 (Jakub Pawlak)

Histogram of Oriented Gradients (HOG) to metoda ekstrakcji cech używana w przetwarzaniu obrazu. Opiera się ona na zliczaniu gradientów zorientowanych w tym samym kierunku, w określonych fragmentach obrazu. Deskryptor HOG opisuje kształt obiektu na obrazie, więc bardzo dobrze nadaje się do zadania rozpoznawania cyfr, ponieważ następuje ono właśnie na podstawie kształtu.

Alorytm najpierw dzieli obraz na komórki o określonym rozmiarze. W przypadku zbioru MNIST użyto komórek 14×14 , uzyskując podział całego obrazu na 4 komórki. W każdej komórce, oblicza się dla każdego piksela lokalny gradient. Następnie, wewnątrz każdej komórki zlicza się gradienty w określonych kierunkach i tworzy się z nich histogram. Aby umożliwić wykrycie linii zarówno ortogonalnych jak i ukośnych, liczba kierunków została ustawiona na 9. W celu poprawy jakości, wartości gradientów są dodatkowo normalizowane w większych grupach. W tym przypadku, użyto grup o rozmiarze 2×2 komórek, co odpowiada całemu obrazowi.

Cały obraz zostaje zatem opisany za pomocą 4 komórek, każda zawierająca wartości dla 9 kierunków. W ten sposób obraz 28×28 zredukowano do 36-elementowego wektora.



Rysunek 4: Przykłady ekstrakcji cech metodą HOG

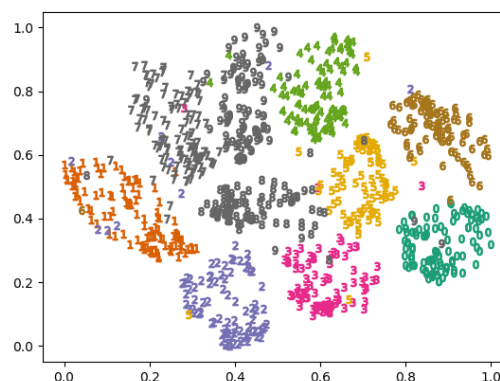
Inne podejścia W tym miejscu zasadnym byłoby wskazać, że w przypadku ekstrakcji cech do 2-elementowego wektora zostały również sprawdzone inne metody. Jednym ze sprawdzonych cech była długość najdłuższej prostej. Obraz był w tym celu najpierw przekształcany na binarny, a następnie poddawany dylatacji. Następnie, sprawdzano wszystkie punkty należące do obrazu parami, i jeżeli prosta pomiędzy nimi była całkowicie zawarta w cyfrze, liczonego odcinka między punktami. Długość najdłuższego odcinka stanowiła wartość jednej z cech. Miało to na celu wyróżnić cyfry z długimi prostymi odcinkami (np. 1, 7), od tych bardziej “krętych” (2, 5, 3).

Drugą cechą była ilość segmentów całkowicie otoczonych przez cyfrę (1 dla cyfr 6,9,0; 2 dla cyfry 8), obliczane za pomocą algorytmu floodfill. Niestety, sprawiało to problemy z powodu dwójakiego sposobu zapisu cyfry 4, jak również istnienia “niedomkniętych” cyfr 9.

Ogółem, udało się uzyskać niezbyt satysfakcjonujące wyniki ok. 30%. Dlatego też, mając na względzie, że posłuży to wizualizacji, postanowiono skorzystać ze statystycznej metody dedykowanej redukcji wymiarów na potrzeby wizualizacji.

t-Distributed Stochastic Neighbor Embedding (t-SNE) jest stochastyczną metodą redukcji wymiarów, często wykorzystywaną przy tworzeniu wizualizacji.

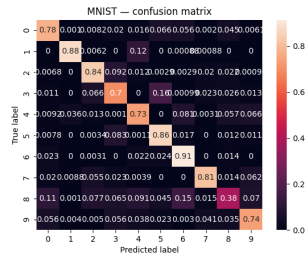
Próbuje ona rozłożyć punkty w przestrzeni docelowej zachowując lokalnych sąsiadów z przestrzeni źródłowej. Metoda ta bada odległości między punktami w przestrzeni źródłowej i przypisuje im rozkład prawdopodobieństwa w oparciu o rozkład standardowy. Następnie wybiera (losowo lub przez PCA) rozkład punktów w przestrzeni docelowej i analizuje ich odległości, przypisując im prawdopodobieństwa oparte o rozkład Cauchy’ego. Następnie za pomocą metody minimalizacji gradientu stara się zminimalizować różnicę pomiędzy rozkładami w przestrzeni źródłowej i docelowej.



Rysunek 5: Wizualizacja docelowej przestrzeni t-SNE

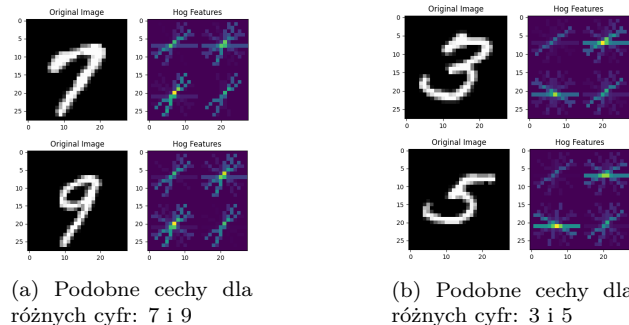
4 Wyniki eksperymentu — Osoba 2 (Jakub Pawlak)

Dla ekstrakcji cech w celu oceny separowalności wytrenowano sieć z małą ilością epok i oceniono macierz pomyłek, która znajduje się na rys 6.



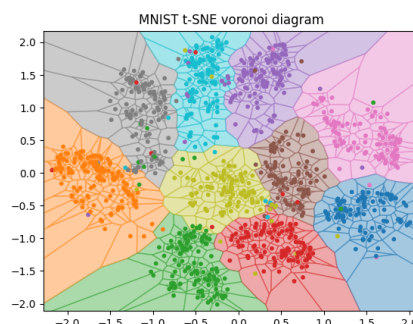
Rysunek 6: Macierz pomyłek na słabo wytrenowanej sieci (ekstakcja HOG)

Zauważyć można, że najgorzej model radzi sobie z rozpoznawaniem cyfr 8,3,4, 9 i 0. Cyfra 8 bardzo często uznawana jest za 0,2,3,4 lub 6. Często mylone są również 3 i 5. Wynika to z faktu, że algorytm HOG wewnątrz komórek uwzględnia tylko ilość gradientów, a nie ich położenie względem siebie. Dla przykładu, na rys. 7 przedstawiono różne cyfry, które po ekstrakcji cech wyglądają podobnie. Tak oto przedstawione cyfry 3 oraz 5 obie posiadają w prawej górnej ćwiartce linię ukośną i poziomą. W cyfrze 3 linia pozioma idzie w prawo, a w cyfrze 5 — w lewo, natomiast dla algorytmu HOG nie ma to znaczenia, ponieważ obie linie znajdują się w tej samej komórce. Podobnie z cyframi 7 i 9, różniącymi się jedynie poziomym domknięciem, co jednak nie znajduje odzwierciedlenia w deskrypcorze HOG, ponieważ pozioma linia już występuje gdzie indziej w tej samej ćwiartce.



Rysunek 7: Różne cyfry, które posiadają podobne cechy

W metodzie t-SNE można ocenić separowalność wizualnie, na podstawie rys. 5, jak również diagramu Woronoja na rys. 8. Od razu widać, że poszczególne klasy są separowalne, choć zdarzają się pola do pomyłek, zwłaszcza przy cyfrach 3,5 i 8, oraz 7,9 i 4.



Rysunek 8: Diagram Woronoja dla zbioru po przekształceniu t-SNE

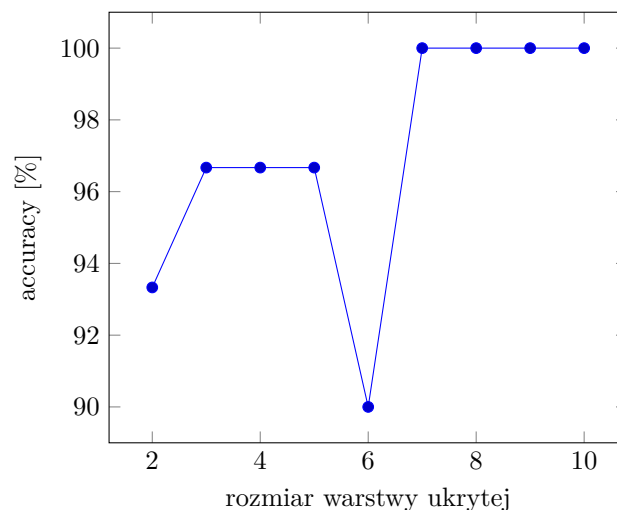
5 Wybór optymalnego modelu

Podczas eksperymentów przeprowadzonych w ramach pierwszego projektu dało się zauważyć, że w miarę zwiększania liczby neuronów w warstwie ukrytej, skuteczność ewentualnie ulegała wypłaszczeniu. W takim przypadku, zwiększanie liczby neuronów skutkowałoby jedynie utrudnieniem obliczeń, bez pozytywnego efektu na skuteczności modelu. Optymalnym modelem jest zatem taki, który maksymalizuje dokładność przy jednoczesnym minimalizowaniu ilości neuronów.

Przeprowadzone eksperymenty nie dostarczyły natomiast jednoznacznego sposobu, aby a priori określić najlepszą liczbę neuronów w warstwie ukrytej. Wybór optymalnego modelu został dokonany poprzez coraz dokładniejsze przeszukiwanie przestrzeni liczb naturalnych, podobne w zasadzie działania do algorytmu *binary search*. Różne wartości liczby neuronów były poddawane procesowi uczenia, którego to wyniki pozwalały na oszacowanie, które wartości są najbardziej obiecujące. Następnie wybierane zostały kolejne ilości neuronów do sprawdzenia — tym razem z sąsiedztwa najlepiej spisujących się w poprzedniej iteracji. Proces powtarzano, aż do znalezienia lokalnego maksimum skuteczności.

Przykładowy przebieg pokazano na rys. 9. Na wykresie pokazano zmianę wartości *accuracy* w zależności od rozmiaru warstwy ukrytej. Najwyższe wartości udało się osiągnąć warstw ukrytych liczących 7 lub więcej neuronów. W takim przypadku zwiększenie liczby neuronów nie skutkuje poprawą skuteczności, więc preferowana jest wartość najmniejsza, t.j. 7.

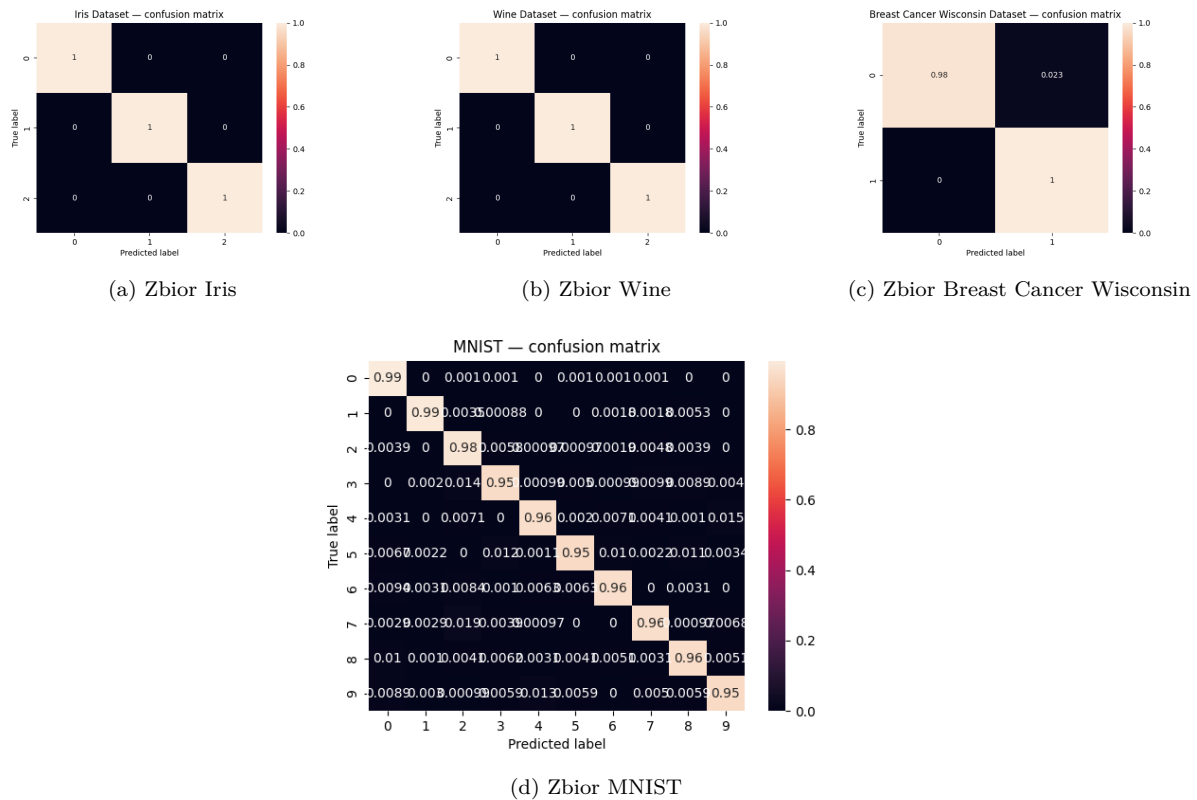
Metoda ta pozwoliła uzyskać satysfakcjonujące wyniki, jednak wskazać należy, że jesteśmy świadomi iż nie daje ona gwarancji, że wybrany model jest globalnie optymalny.



Rysunek 9: Przykładowe wartości *accuracy* dla różnych rozmiarów warstwy ukrytej na zbiorze Iris

6 Wyniki klasyfikacji dla pierwszego sposobu ekstrakcji cech

Poniżej przedstawiono macierze pomyłek i wartości accuracy dla pozostałych zbiorów danych, oraz dla pierwszej metody ekstrakcji cech ze zbioru MNIST. Wszystkie sieci wykorzystują podobną architekturę, składającą się z warstwy wejściowej, jednej warstwy ukrytej i warstwy wyjściowej. Rozmiar warstwy wejściowej i wyjściowej determinowany jest charakterystyką zbioru danych — odpowiednio: ilością cech i ilością klas. Rozmiar warstwy ukrytej został ustalony metodą opisaną na str. 5. Wybrano możliwie małą wartość tak, aby jej zwiększenie nie powodowało poprawy skuteczności modelu. Dodatkowo, w warstwie ukrytej aplikowana jest funkcja ReLU.



Rysunek 10: Wyniki klasyfikacji dla 1 sposobu ekstrakcji cech

Zbiór danych	Wartość Accuracy	Rozmiar warstwy ukrytej
Iris	100%	7
Wine	100%	2
Breast Cancer Wisconsin	99.12%	3
MNIST	97.16%	64

Tablica 1: Wartości accuracy wytrenowanego modelu

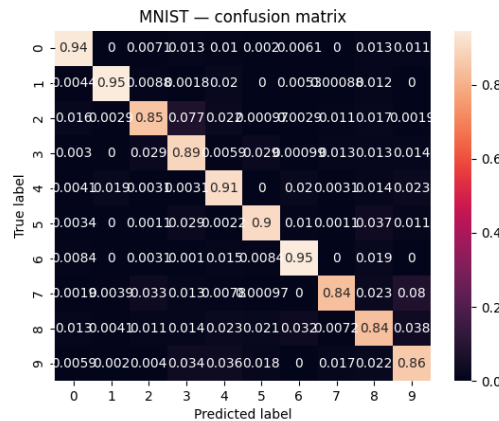
Z wyników widać, że dla zbiorów Iris i Wine udało się uzyskać stuprocentową skuteczność. Dziwne wydaje się, że dla zbioru Iris konieczne było użycie warstwy ukrytej większej od zarówno warstwy wejściowej jak i wyjściowej. Dla zbioru Breast Cancer Wisconsin uzyskano bardzo satysfakcjonujące 99.12% accuracy. Wynik 97% dla zbioru mnist nie jest zadziwiający, biorąc pod uwagę, że niektóre cyfry są trudne do rozróżnienia nawet dla człowieka. Nie jest zadziwiająca również najlepsza ilość neuronów w warstwie ukrytej, biorąc pod uwagę fakt, że warstwa wejściowa liczy ich 784.

7 Wyniki klasyfikacji — Osoba 1 (Magdalena Pakuła)

8 Wyniki klasyfikacji — Osoba 2 (Jakub Pawlak)

HOG

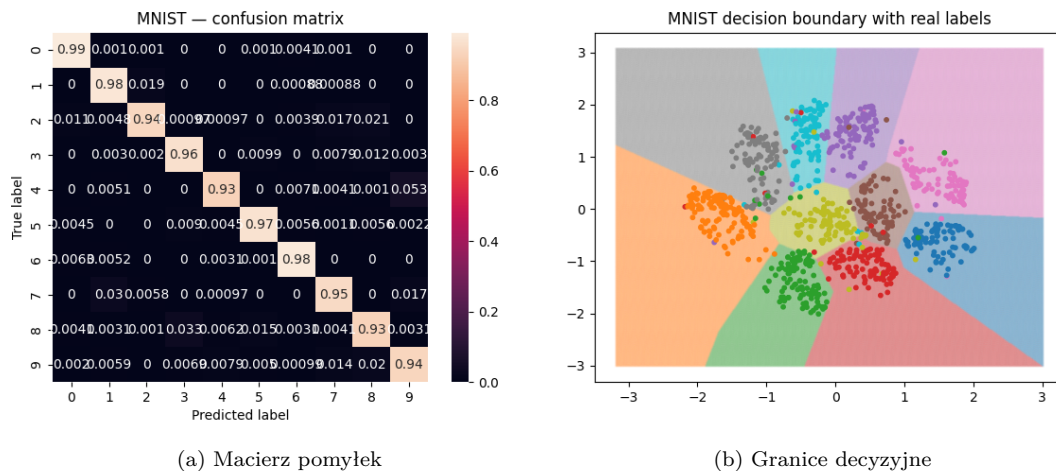
Wyniki klasyfikacji z metodą HOG przedstawiono na rys. 11. Widać znaczącą poprawę w stosunku do wstępnych oczekiwań, choć tak jak przewidziano, z cyframi 2,3,7,8,9 model radzi sobie słabiej niż z resztą. Tak jak przewidywano, często mylone są cyfry pary (2,3), (3,5), (7,9), (8,6), (8,9). Ogólna wartość accuracy osiągnęła 89.21%.



Rysunek 11: Wyniki klasyfikacji dla ekstrakcji cech metodą HOG (Accuracy: 89.21%).

t-SNE

Na rys. 12a przedstawiono macierz pomyłek oraz przebieg granicy decyzyjnej. Model poradził sobie bardzo dobrze, jednak można zauważyć drobne pomyłki. Zgodnie z przewidywaniami, 4 i 7 były często uznawane za 9, jak również mylone były cyfry z grupy 3,5,8. Pomimo mniejszej liczby cech, model osiągnął lepszą niż w poprzedniej metodzie wartość accuracy wynoszącą 95.67%. Wskazać jednak należy istotną wadę tej metody, mianowicie t-SNE operuje na całym zbiorze danych, więc użycie tej metody ekstrakcji cech nie będzie można zastosować w przypadku pojawienia się nowych danych.



(a) Macierz pomyłek

(b) Granice decyzyjne

Rysunek 12: Wyniki klasyfikacji dla ekstrakcji cech metodą t-SNE (Accuracy: 95.67%)

Z obydwu eksperymentów dojść można do wniosku, że ręczna ekstrakcja cech nie jest zadaniem łatwym, ponieważ w obu przypadkach poskutkowało ona pogorszeniem skuteczności modelu.