

Projekt 1 - Sieć Hopfielda - Raport

1. Opis rozwiązania

Sieć Hopfielda została po raz pierwszy pokazana przez Johna Hopfielda w 1982 roku. Jest to rodzaj rekurencyjnej sieci neuronowej, stan każdego neuronu zależy od stanu neuronów w poprzedniej jednostce czasu. W tym przypadku, każdy neuron jest połączony z resztą neuronów w sieci, połączenia między nimi mogą być reprezentowane jako macierz wag W_{nn} :

- połączenia między neuronami są symetryczne, $W = W^T$,
- żaden neuron nie jest połączony z samym sobą, $W_{ii} = 0$,
- w klasycznej sieci hopfielda, neurony przyjmują stany binarne -1 lub 1.

Sieć Hopfielda należy do sieci asocjacyjnych, potrafi odtwarzać "zapamiętane" wzorce na podstawie niepełnych lub zaszumionych danych.

Do uczenia sieci stosuje się metodę opartą na regule Hebb'a:

$$W_{ij} = \frac{1}{p} \sum_{k=1}^p x_i^{(k)} x_j^{(k)}$$

W praktyce jest to średnia iloczynów zewnętrznych każdego ze wzorców.

Liczba poprawnie pamiętanych wzorców jest ograniczona. Uczenie hebbowskie daje niską pojemność pamięci i żeby utrzymać liczbę błędów na poziomie poniżej 0.37%, maksymalna liczba wzorców to: $p_{max} = 0.138x$, gdzie x jest liczbą neuronów.

W praktyce pojemność wynosi:

$$p_{max} = \frac{x}{2\log_2(x)}$$

Po treningu sieć może odtwarzać zapamiętane wzorce na podstawie wzorca wejściowego $x(0)$. W każdej iteracji stany neuronów aktualizowane są na podstawie znaku aktywacji:

$$y_i(t + 1) = \operatorname{sgn}\left(\sum_{j=1}^n W_{ij} y_j\right),$$

przy czym w aplikacji przyjęto konwencję, że gdy aktywacja jest równa 0, stan neuronu zostaje ustawiony na 1.

Neurony mogą być aktualizowane w dwóch trybach:

- asynchronicznie - w losowej kolejności - co zapobiega sytuacji że neurony oscylują między dwoma stanami,
- synchronicznie - wszystkie jednocześnie - łatwiejsze do zrównoleglenia.

Sieć dąży do minimalizacji funkcji energii:

$$E = - \sum_{i,j,i < j}^n W_{ij} y_i y_j - \sum_{i=0}^n b_i y_i ,$$

przy czym b jest wektorem wyrazów wolnych - średnią wektorów zapamiętanych wzorców.

Energia maleje podczas procesu odtwarzania, aż sieć osiągnie stan stabilny.

Jednym z głównych zastosowań sieci Hopfielda jest odtwarzanie zaszumionych lub częściowo zakrytych binarnych obrazów. Obraz taki jest "spłaszczany" do wektora n-elementowego. Inne zastosowania to rozwiązywanie NP-trudnych problemów kombinatorycznych, np. problemu komiwojażera oraz ogólna korekcja błędów.

2. Zależności aplikacji

Aplikacja działa w Pythonie w wersji 3.12.3. Wymagane są również następujące pakiety pip:

- PyQt6 (6.9.1)
- numpy (2.3.2)
- pillow (11.3.0)
- datasets (4.0.0)
- matplotlib (3.10.6)

oraz wszelkie ich zależności. Pełną listę pakietów można znaleźć w pliku requirements.txt, w głównym katalogu repozytorium. Można zainstalować wszystkie pakiety używając polecenia:

```
pip install -r requirements.txt
```

Do funkcji wczytania wzorców MNIST wymagane jest połączenie sieciowe.

Aby uruchomić aplikację wystarczy uruchomić poprzez pythona skrypt app.py, w folderze app repozytorium:

- Windows:

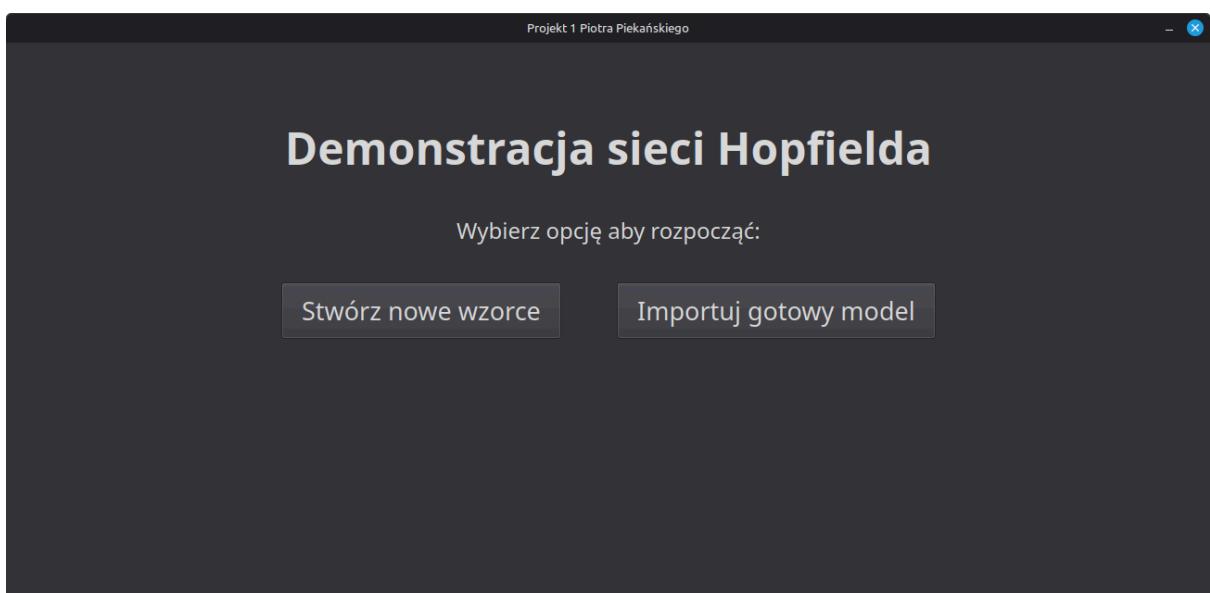
```
python ./app/app.py
```

- Linux:

```
python3 ./app/app.py
```

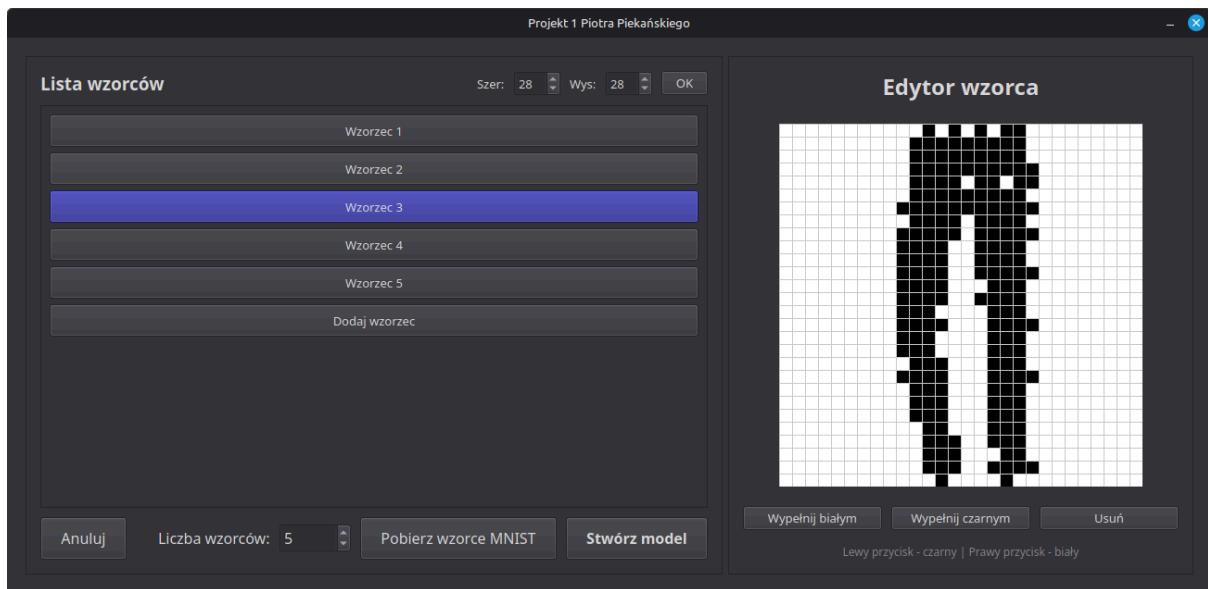
3. Funkcje programu

Aplikacja graficzna demonstruje sieć Hopfielda na przykładzie rekonstrukcji (odtwarzania) czarno-białych obrazków. Użytkownik ma możliwość własnoręcznego rysowania obrazków - wzorców lub wczytania istniejących wzorców z zestawu danych MNIST Fashion. Po utworzeniu modelu użytkownik może spróbować odtworzyć jeden ze wzorców na bazie obrazka wejściowego, zobaczyć przebieg odtwarzania wzorca oraz funkcję energii.



Rysunek 3.1. Widok startowy programu

Po uruchomieniu programu widoczne jest menu główne (Rys. 3.1.), z którego można przejść do trybu tworzenia wzorców (Rys. 3.2.), lub wybrać już przygotowane wzorce i modele w postaci pliku pickle (rozszerzenie .pkl), po wybraniu poprawnego pliku, aplikacja przenosi użytkownika do trybu odtwarzania wzorca (Rys. 3.3.)



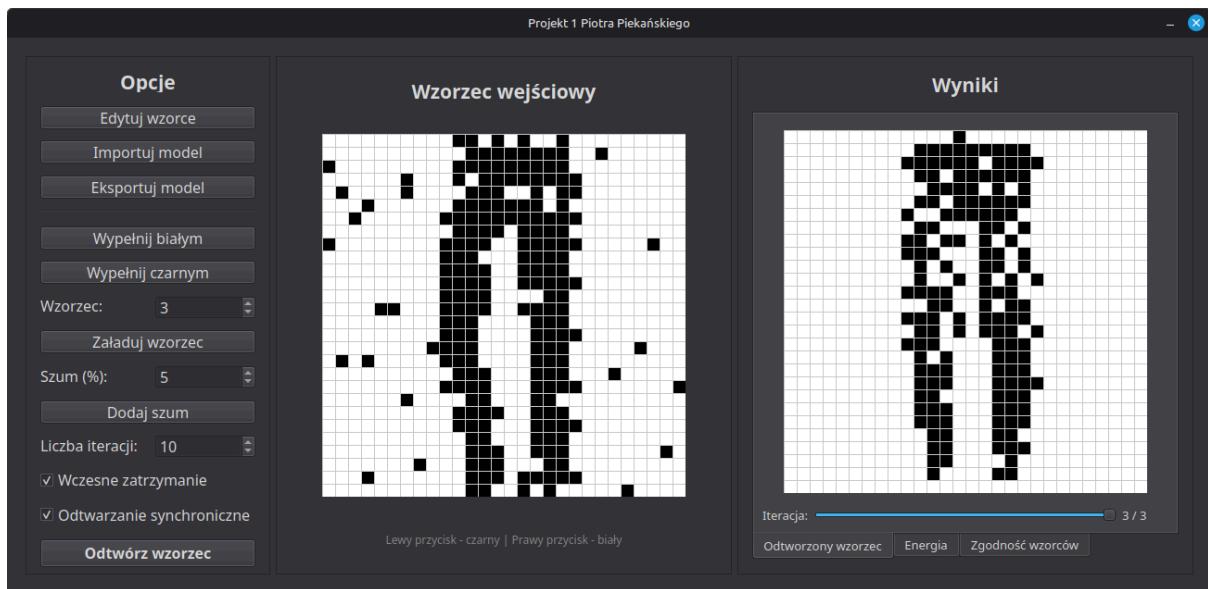
Rysunek 3.2. Widok tworzenia wzorców do zapamiętania przez sieć

Widok tworzenia wzorców (Rys. 3.2.) składa się z listy wzorców po lewej oraz edytora wzorca po prawej. Po wejściu z menu głównego, lista wzorców zawiera tylko jeden pusty (biały) wzorzec. Kliknięcie przycisku “Dodaj wzorzec” tworzy nowy obrazek i rozpoczyna jego edycję. W każdej chwili można powrócić do edycji dowolnego obrazka.

Powyżej listy znajdują się dwa pola numeryczne z wymiarami obrazka. Po zatwierdzeniu nowych wymiarów, czyszczona jest lista wzorców (nie ma potwierdzenia) oraz tworzona jest nowa plansza do rysowania.

Rysowanie obrazków odbywa się za pomocą myszy, w stylu programów graficznych, jak MS Paint. Lewy przycisk myszy rysuje czarnym kolorem, a prawy białym. Biały i czarny odpowiadają odpowiednio stanom -1 i 1 neuronów. Pod planszą są przyciski wypełniające planszę jednym kolorem oraz przycisk usuwający wzorzec z listy. Zmiany dokonane w obrazku są natychmiast zapamiętywane w liście.

Pod listą znajduje się przycisk “Anuluj”, cofający zmiany i wracający do poprzedniego widoku (bez potwierdzenia). “Pobierz wzorce MNIST” zastępuje listę określoną liczbą wzorców z MNIST Fashion (pobieranie może potrwać dość długo, zalecany jest zapis modelu). Wzorce są automatycznie skalowane do wybranych wymiarów i binaryzowane. Przycisk “Stwórz model”, trenuje sieć hopfielda na podstawie wzorców oraz przechodzi do interfejsu odtwarzania (Rys. 3.3.).



Rysunek 3.3. Widok odtwarzania zapamiętanego wzorca

Widok odtwarzania zapamiętanych wzorców (Rys. 3.3.) stanowi główny element aplikacji. Pozwala na przetestowanie wytrenowanej sieci.

Panel po lewej stronie zawiera wiele opcji:

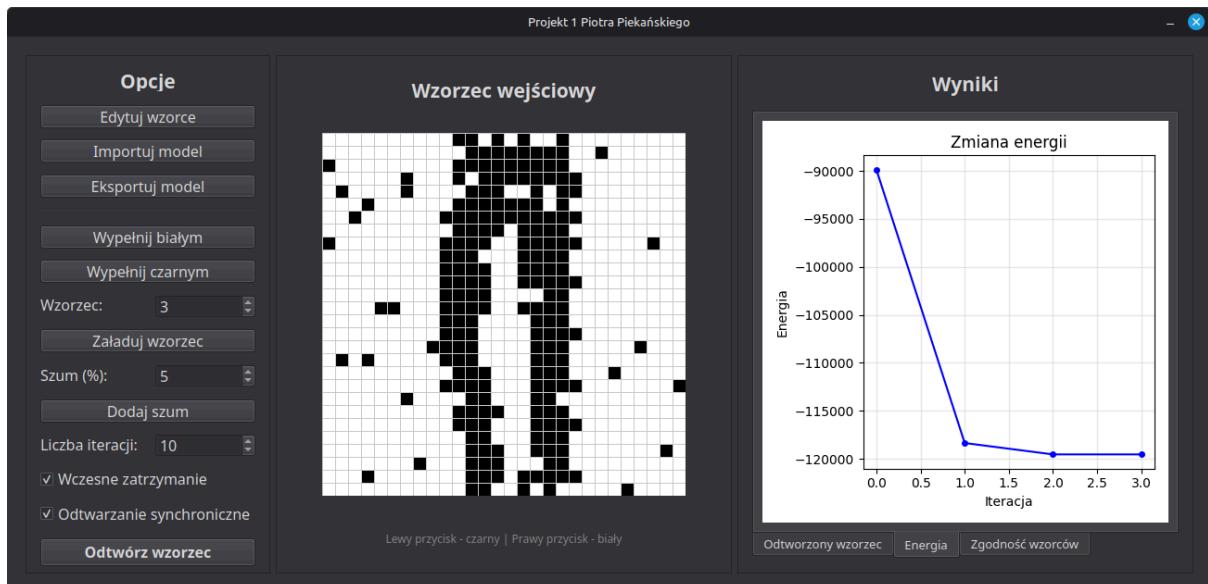
- “Edytuj wzorce” - umożliwia powrót do interfejsu edycji wzorców i ponowne wytrenowanie modelu,
- przyciski do zapisu i odczytu modelu z pliku pickle, jak w menu głównym (Rys. 3.1.),
- przyciski wypełniania białym i czarnym, jak w interfejsie edycji wzorców,
- “Załaduj wzorzec” kopiuje wybrany wzorzec do panelu rysowania,
- “Dodaj szum”, odwraca kolor określonej części pikseli w procentach,
- pole numeryczne z maksymalną liczbą iteracji odtwarzania,
- “Wczesne zatrzymanie” - przerywa odtwarzanie gdy wartość funkcji energii nie zmieni się po jednej iteracji,
- pole wyboru trybu synchronicznego lub asynchronicznego odtwarzania wzorca,
- “Odtwórz wzorzec” uruchamia proces rekonstrukcji dla wzorca wejściowego.

Rysowanie obrazka w środkowym panelu odbywa się identycznie jak w tworzeniu wzorców do modelu.

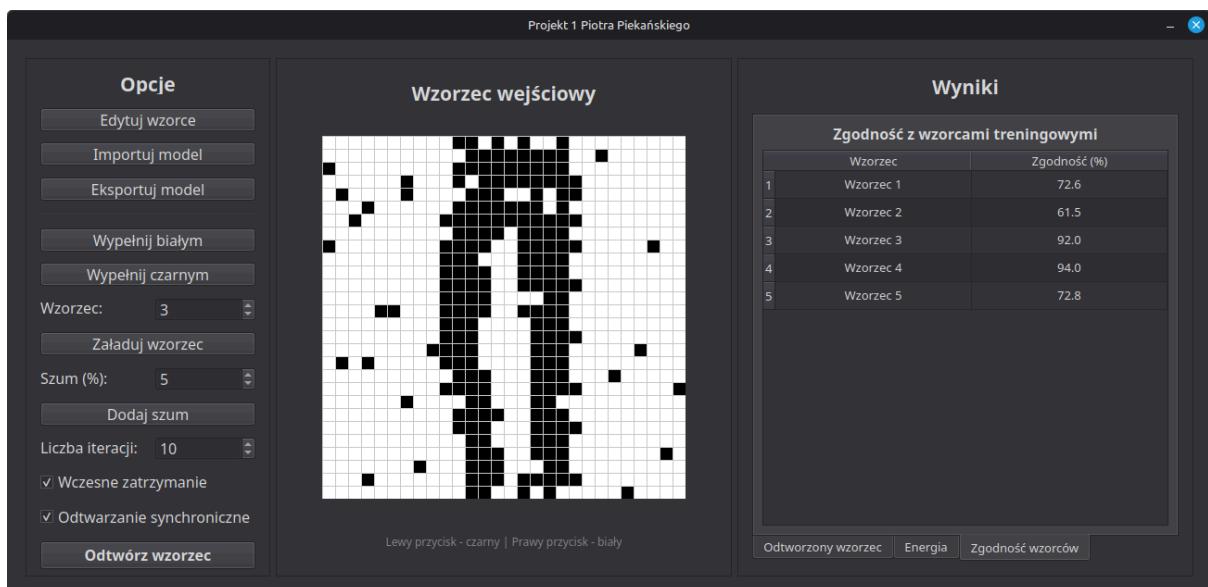
Rekonstrukcja trwa, dopóki różnica energii w dwóch kolejnych iteracjach nie stanie się bliska零. Jest również gorny limit 100 iteracji. Po zakończeniu, w panelu po

prawej ukazuje się odtworzony wzorzec, suwakiem poniżej można spojrzeć na stan obrazka w dowolnej iteracji, lub po każdej zmianie neuronów w przypadku odtwarzania asynchronicznego.

W miejscu obrazu wynikowego można pokazać wykres energii sieci (Rys. 3.4.) lub tabelę przedstawiającą zgodność wynikowego obrazu z wzorcami zapamiętanymi przez sieć (procent pikseli, które są takie same u obu obrazów) (Rys. 3.5.).



Rysunek 3.4. Poprzedni widok, z włączoną zakładką z funkcją energii



Rysunek 3.5. Poprzedni widok, z włączoną zakładką ze zgodnością obrazów

4. Symulacje i wyniki

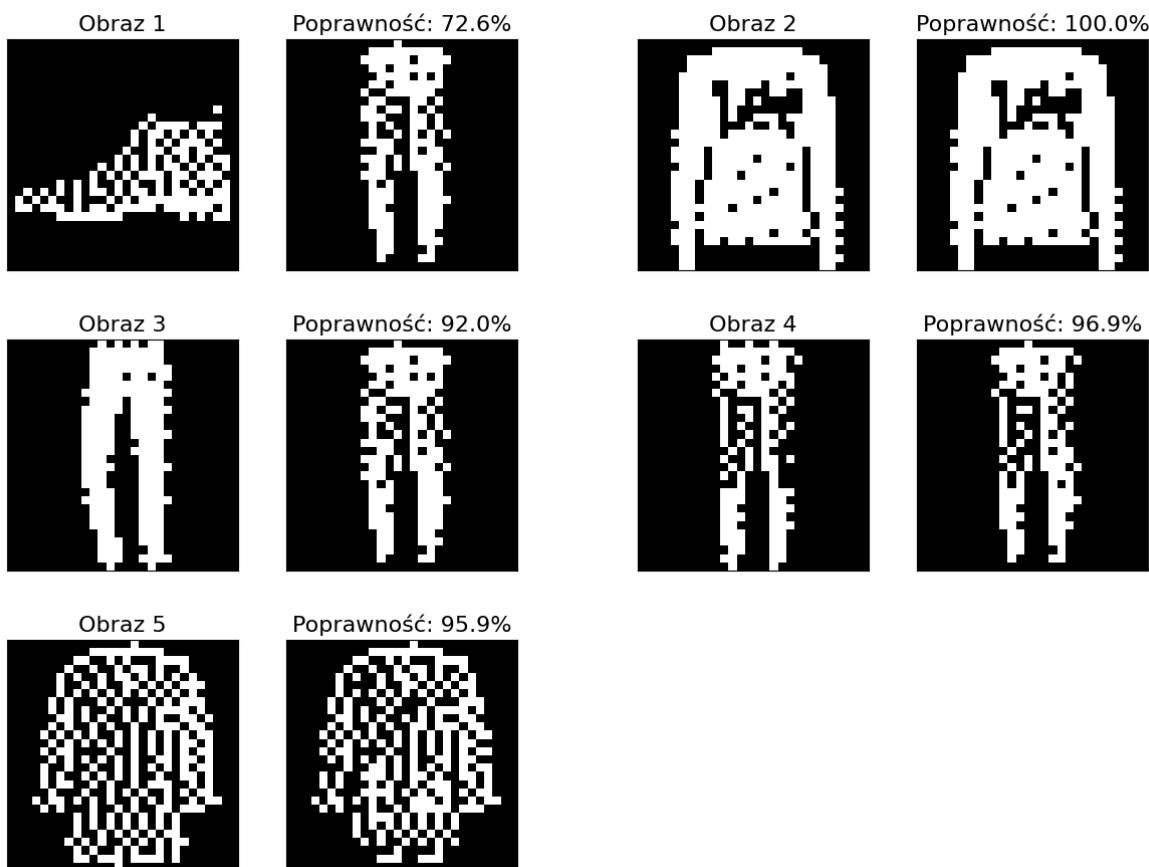
Przeprowadzono testy na pięciu obrazach ze zbioru danych fashion MNIST, które zostały sprowadzone do postaci binarnej. Zachowano oryginalny rozmiar 28x28 pikseli, co za tym idzie sieć składa się z 784 neuronów. Ograniczenie maksymalnej liczby wzorców wynosi więc:

$$p_{max} = \frac{784}{2\log_2(784)} \approx 40$$

Zbiór 5 wzorców spełnia ten limit.

Aby polepszyć odtwarzalność wyników użyto synchronicznego procesu odtwarzania wzorców. Odtwarzanie kończono po stabilizacji poziomu energii (dalejsze iteracje nie zmieniały stanu sieci).

Najpierw przeprowadzono próbę kontrolną - wprowadzono do wytrenowanej sieci oryginalne wzorce (Rys. 4.1).



Rysunek 4.1. Wyniki odtworzenia oryginalnych obrazów.

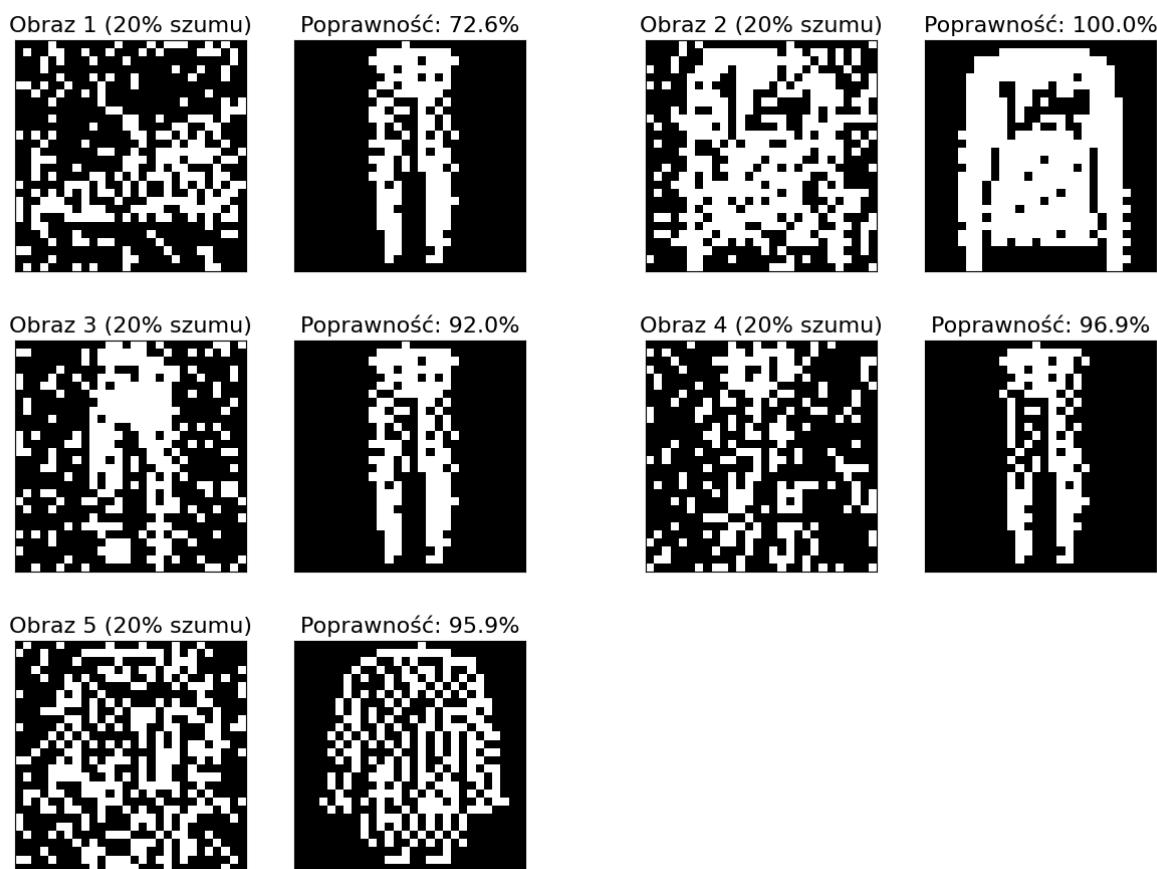
Obraz 1 (but) - został błędnie odtworzony jako obraz przypominający spodnie (obraz 3). To prawdopodobnie wynika z faktu, że obrazy 3 i 4 (oba przedstawiające spodnie) są bardzo podobne do siebie i zostały użyte podczas treningu.

Obrazy 3 i 4 - ze względu na wysokie podobieństwo między sobą, sieć "pomieszała" je, tworząc hybrydę zamiast odtworzyć oryginalne obrazy.

Obraz 2 - pozostał niezmieniony po procesie odtwarzania.

Obraz 5 - zachował kształt zbliżony do oryginału z drobnymi modyfikacjami.

Następnie przeprowadzono próbę odtworzenia tych samych obrazów, po nadaniu im szumu 20% (Rys. 4.2.).



Rysunek 4.2. Wyniki odtworzenia zaszumionych obrazów

Wzorce zostały odtworzone do identycznej postaci jak w eksperymencie kontrolnym (bez szumu) (Rys. 4.1.). Pomimo błędów opisanych wcześniej, wyraźnie widać znaczące możliwości sieci w zakresie usuwania szumu z wzorców wejściowych.