

Patryk Orzechowski\*, Zbigniew Mikrut\*

## Strategie poprawy efektywności uczenia sieci neuronowej\*\*

### 1. Wprowadzenie

Rozpoznawanie ręcznie pisanych znaków jest sztandarowym problemem klasyfikacyjnym stawianym przed sieciami neuronowymi (por. [1, 2, 4, 9, 11]). Zdolności generalizacji sieci pozwalają na rozróżnianie przez sieć całkowicie nowych wzorców na podstawie zgromadzonego wcześniej doświadczenia w procesie uczenia, dzięki czemu sieci neuronowe stanowią obecnie jeden z efektywniejszych klasyfikatorów (por. [3]). Ich efektywność zależy od wielu czynników, wśród których ważną rolę odgrywa sposób uczenia.

Jako przykład do sprawdzenia wpływu metody uczenia na wyniki rozpoznawania wybrano problem rozpoznawania ręcznie pisanych cyfr przez sieć neuronową ze wsteczną propagacją błędów, w przypadku uczenia metodą sekwencyjną. Sieć potraktowano tutaj jako klasyczną „czarną skrzynkę”, na której wejścia podawano wybrane sekwencje obrazów cyfr. Na podstawie obserwacji wyjść sieci neuronowej próbowano wyciągnąć wnioski o działających w jej wnętrzu mechanizmach oraz starano się określić jakie informacje są potrzebne sieci, aby uczenie było efektywniejsze, a zdolności do generalizacji większe. Do celów badawczych użyto bazy obrazów ręcznie pisanych cyfr zgromadzoną w Laboratorium Biocybernetyki AGH. Baza ta jest znacznie mniejsza od popularnej bazy cyfr MNIST (zmodyfikowana baza National Institute of Standards and Technology (NIST) z 60 tys. cyfr w zbiorze uczącym oraz 10 tys. w zbiorze testowym, por. [3, 6]). Wykorzystywana baza liczy 2000 obrazów ręcznie pisanych cyfr w zbiorze uczącym oraz tyle samo w zbiorze testowym. Cyfry zapisane są w skali odcieni szarości, w różnej rozdzielczości.

Przyjęto kryterium decydujące o jakości rozpoznania jako różnicę pomiędzy dwiema wartościami wyjść sieci: wartości odpowiadającej prawidłowemu rozpoznaniu cyfry oraz najsilniej pobudzonemu neuronowi wyjściowemu spośród rozpoznań błędnych (por. [9]). Łatwo zauważyć, że w przypadku pozytywnego rozpoznania danej cyfry wskaźnik ten jest dodatni i pokazuje, jak dalece sieć jest przekonana co do prawidłowości swojej odpowiedzi. Błędne rozpoznanie sprawia, że wskaźnik przyjmuje wartość ujemną. Im silniej sieć przekonana jest, że podana przez nią odpowiedź jest właściwa, tym wartość bezwzględna wskaźnika jest większa.

---

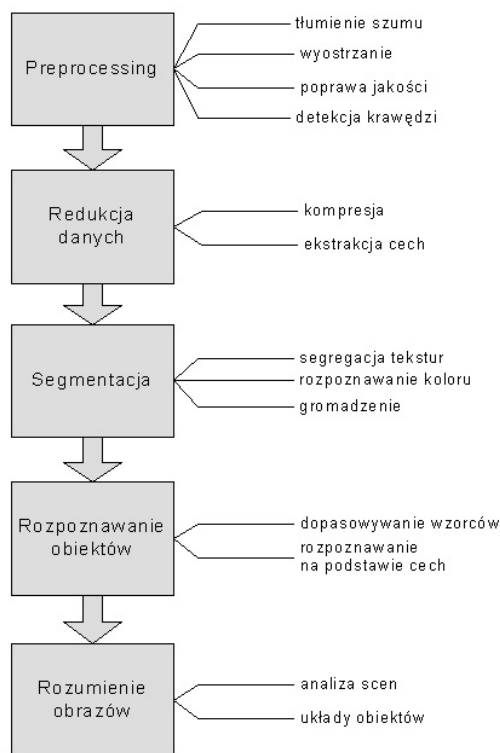
\* Katedra Automatyki, Akademia Górniczo-Hutnicza w Krakowie

\*\* Praca częściowo finansowana z badań własnych – nr AGH 10.10.120.39

Do badań wybrano eksperymentalnie architekturę sieci 216-22-10, z sigmoidalną funkcją przejścia w warstwie wyjściowej oraz tangensoidalną w warstwie ukrytej. Środowiskiem programistycznym do przeprowadzania eksperymentów był MATLAB w wersji 7.1.0.246 (R14SP3): użyto biblioteki do przetwarzania obrazów (*Image Processing Toolbox 5.1*), jak również biblioteki z sieciami neuronowymi (*Neural Network Toolbox 4.0.6*).

## 2. Przetwarzanie wstępne

Klasyczny schemat postępowania w analizie i przetwarzaniu obrazu [1, 5, 11] wyodrębnia szereg czynności poprzedzających właściwe rozpoznawanie, obejmujących przetwarzanie wstępne (służące poprawie jakości obrazów wzorcowych oraz eliminacji zakłóceń), redukcję wymiarowości oraz segmentację. Mają one na celu wyodrębnienie zespołu cech, na podstawie których przeprowadzone zostanie rozpoznanie – por. rysunek 1.

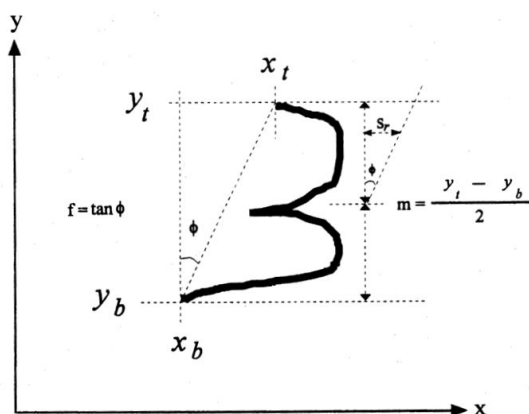


Rys. 1. Klasyczny schemat analizy i przetwarzania obrazu (wg [1])

Zastosowano kilkietapowy preprocessing, składający się m.in. z filtracji medianowej, wyodrębniania cyfry, szkieletyzacji oraz dylatacji – w celu wzmocnienia kształtu cyfry. Właściwe eksperymenty przeprowadzono na obrazach przeskalowanych do rozmiaru 18×12 pikseli.

## 2.1. Korekcja nachylenia

Problemy z rozpoznawaniem pochylego pisma i celowość stosowania korekcji nachylenia omawiane są w wielu pracach (m.in. [4] czy [11]), Prezentowany algorytm korekcji nachylenia cyfr powstał jako modyfikacja algorytmów użytych w [3] oraz [8].



Rys. 2. Korekcja nachylenia (na podstawie [8])

Korekcję nachylenia dokonuje się poprzez adaptację algorytmu z [8], w którym kąt nachylenia  $f$  wyznacza się na podstawie informacji o pierwszym niezerowym lewym górnym  $(x_t, y_t)$  i lewym dolnym  $(x_b, y_b)$  pikselu (por. rys. 2), wg wzoru

$$f = \frac{x_t - x_b}{y_b - y_t} \quad (1)$$

Następnie dla każdego wiersza  $r = 1, 2, \dots$  obliczana jest współrzędna przesunięcia według wzoru

$$s_r = (r - m) \cdot f \quad (2)$$

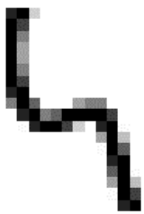


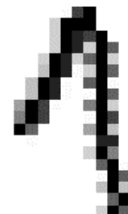
gdzie:

- $r$  – indeks wiersza obrazu,
- $m$  – środek obrazu w pionie.

Wykorzystanie informacji pochodzącej wyłącznie z konturu cyfr dawałoby sporą przestrzeń do potencjalnych przekłamań, postanowiono więc opracować skuteczniejsze kryterium do wyznaczania stopnia pochyłości cyfry. Rozwiązanie, które zaadaptowano z pracy [3], polegało na wyznaczeniu dwóch obszarów, obejmujących odpowiednio 25% górnej oraz 25% dolnej części obrazu, oraz wyznaczenia w każdym z nich centrów grawitacji. Poprzez porównanie współrzędnych centrów grawitacji wyznaczano kąt nachylenia cyfry, a następnie dokonywano „prostowania” cyfry, polegającego na wyznaczeniu dla każdego wiersza wartości przesunięcia w porównaniu z obrazem pierwotnym.





Efekt działania algorytmu dla przykładowych danych przedstawia tablica 1.

**Tablica 1**  
Przykładowe cyfry przed i po korekcji nachylenia

Przed	Po	Przed	Po
			

Ciekawą obserwacją jest wpływ korekcji nachylenia pojedynczych matryc z cyframi na obrazy uśrednionych cyfr danego rodzaju. Zauważyć można, że średnie obrazy cyfr po „wprostowaniu” są dużo wyraźniejsze, zwłaszcza w górnym oraz dolnym obszarze (por. tablica 2). W wyniku tej korekty „cechy cyfr”, czyli piksele, dzięki którym sieć rozpoznaje daną cyfrę, grupują się dla poszczególnych cyfr w sąsiednich sektorach.

**Tablica 2**  
Uśrednione obrazy cyfr przed oraz po korekcji nachylenia

Przed	Po
	
	

Konsekwencją zastosowania przetwarzania wstępnego jest zwiększenie skuteczności rozpoznawania cyfr dla wszystkich omawianych dalej metod o ok. 5% (z 86% do ok. 91%).

### 3. Strategie poprawiające efektywność

Jak wskazuje praca [10], sam sposób prezentacji wzorców sieci neuronowej może znacząco wpływać na jej szybkość uczenia oraz późniejszą jakość rozpoznawania. Nie istnieją

jednak gotowe schematy postępowania, które gwarantowałyby lepszą skuteczność. W pracy [7] zaprezentowano kilka ciekawych rozwiązań, które prowadzić mają do szybszego uczenia się sieci neuronowej z wsteczną propagacją błędów. Przedstawione pomysły zainspirowały niektóre eksperymenty, opisane w dalszej części tego rozdziału.

Proces uczenia może przebiegać na dwa zasadnicze sposoby [7]. W pierwszym z nich (tzw. uczenie batchowe; *batch learning*) aktualizacja wag sieci neuronowej następuje po pełnym przeglądnięciu zbioru uczącego. Kolejność pokazywania poszczególnych egzemplarzy sieci nie ma żadnego znaczenia. Drugi sposób (tzw. uczenie sekwencyjne, *uczenie on-line*) różni się tym, że aktualizacja wag następuje po każdorazowej prezentacji wzorca. Pojedyncza iteracja algorytmu uczenia to wybór wektora wejściowego według określonego schematu, zaprezentowanie go sieci, po czym następuje modyfikacja wag sieci.

Aby odzwierciedlić przebieg procesu uczenia, po każdej iteracji algorytmu sieć testowano za pomocą zbioru uczącego oraz zapisywano do osobnego pliku parametry takie jak: skuteczność rozpoznawania zbioru uczącego, odpowiedź sieci na egzemplarze cyfr oraz poziom poszczególnych wyjść sieci (determinujących pewność rozpoznania). W trakcie procesu uczenia obserwowano zarówno skuteczność rozpoznania, jak również pewność klasyfikacji poszczególnych wzorców, zarówno w przypadku zbioru uczącego, jak i testowego. Aby sieci nie zostały „przeuczone”, przerywano proces uczenia po 200 epokach.

### 3.1. Uczenie batchowe

Zaletami uczenia batchowego są przede wszystkim dobrze poznane warunki zbieżności oraz łatwiejsza analiza teoretyczna dynamiki wag oraz zbieżności. Metoda prowadzi w stronę minimum po najbardziej stromym gradiencie.

Użycie metody batchowej do przetwarzania dużych zbiorów uczących, zawierających częstokroć nadmiarowe kopie wektorów wejściowych, jest jednak mniej efektywne. Przykładowo, dla zbioru składającego się z 1000 wzorców, w którym występuje 10 identycznych kopii 100-elementowego zbioru, metodzie sekwencyjnej wystarczy pokazać jedynie 100 wzorców, podczas gdy batchowa wymaga przejrzania całego zbioru.

### 3.2. Uczenie sekwencyjne losowe

Sztandarowym algorytmem sekwencyjnego uczenia sieci neuronowej jest prezentacja cyfr w kolejności losowej. W każdej epoce uczenia dokonywana jest permutacja zbioru uczącego, wagi w sieci aktualizowane są w procesie uczenia po każdorazowym zaprezentowaniu sieci wzorca.

### 3.3. Uczenie najgorzej rozpoznawanymi cyframi

Podstawą drugiego algorytmu była hipoteza, że sieć będzie lepiej uczyła się, jeśli pokazywane jej będą tylko te wzorce cyfr, których sieć nauczyła się najgorzej. W każdej iteracji algorytmu, przed właściwym procesem uczenia dokonywano więc rozpoznawania całego zbioru treningowego, w celu wyłonienia najgorzej rozpoznawanych egzemplarzy cyfr. Egzemplarze te następnie umieszczano w nowym zbiorze uczącym (będącym podzbiorem całego zbioru treningowego), który to zbiór wykorzystywano następnie do nauczania sieci.

Zastosowano równoliczną reprezentację poszczególnych cyfr w zbiorze uczącym, w celu uniknięcia rozregulowania sieci poprzez prezentację wielu tych samych cyfr pod rząd. Efektem ubocznym uczenia najgorzej rozpoznanymi cyframi może być pogorszenia rozpoznawania egzemplarzy pierwotnie lepiej rozpoznawanych, co ma miejsce w przypadku zbyt częstego pokazywania sieci najgorzej rozpoznanych wzorców [7].

### 3.4. Wykorzystanie matrycy pomyłek

Matryca pomyłek (por. rys. 3) przedstawia ilościowo, które cyfry z którymi zostały pomyłone w procesie rozpoznawania. Na przekątnej matrycy znajdują się rozpoznania prawidłowe, zaś we wszystkich polach poza przekątną – liczba egzemplarzy cyfr pomyłonych przez sieć z innymi w procesie rozpoznawania (kolorem szarym zaznaczono pomyłki).

		Rozpoznane znaki									
		'0'	'1'	'2'	'3'	'4'	'5'	'6'	'7'	'8'	'9'
Pokazywane znaki	'0'	197	0	0	1	1	0	0	0	0	1
	'1'	0	188	0	2	1	2	2	2	0	3
	'2'	2	1	191	1	0	0	2	0	3	0
	'3'	0	2	1	179	1	2	0	3	3	9
	'4'	2	15	0	1	166	1	5	6	2	2
	'5'	4	12	0	40	6	76	10	5	18	29
	'6'	4	0	1	0	5	2	188	0	0	0
	'7'	0	2	0	1	2	0	0	191	1	3
	'8'	2	5	3	0	0	2	1	1	174	12
	'9'	4	6	1	14	2	2	0	3	16	152

Rys. 3. Przykładowa matryca pomyłek  
Objaśnienia w tekście

Dokładniejsza analiza przebiegu zmian matrycy pomyłek w procesie uczenia sieci (a zwłaszcza zmian wartości kilku maksimów w matrycy, odpowiadających błędnym rozpoznaniom) wykazała, że największe wartości błędnych rozpoznań (kolor ciemnoszary na rys. 3) znikają najwolniej w procesie uczenia sieci, przez co to głównie one generują błędy w klasyfikacji.

Do zbioru uczącego postanowiono więc oddelegowywać w każdej iteracji wszystkie egzemplarze cyfr należących do „błędnych maksimów” oraz egzemplarze tych cyfr, z którymi były one mylone („symetryczne pomyłki” – kolor szary). Całość dopełniano egzemplarzami cyfr, które były najgorzej rozpoznane. Zdecydowano, że obserwacja zmian roz-

pocznie się w momencie częściowego wykształcenia się wag w sieci neuronowej. Postawiono zastosować krótki „rozbieg” przed rozpoczęciem obserwacji zmian – początkowo 20 iteracji przeprowadzono wg algorytmu losowego; podobnie czyniono także w przypadku w pełni poprawnego rozpoznania zbioru uczącego.

#### 4. Eksperymenty i wyniki

Wyniki poszczególnych eksperymentów pokazuje tabela 1, w której zestawiono skuteczności rozpoznania zbioru uczącego oraz testowego wraz ze średnią pewnością rozpoznania cyfr ze zbioru uczącego w 200 iteracji algorytmu. Proces uczenia celowo przerwano wcześniej – aby nie dopuścić do przeuczenia sieci, czyli nadmiernego dopasowania się do wzorców. Przeuczenie sieci prowadzi do znacznego pogorszenia zdolności generalizacyjnych sieci i w efekcie spadku skuteczności rozpoznawania [5].

**Tabela 1**

Skuteczność oraz pewność rozpoznania sieci neuronowych. Uczenie przerwano po 200 epokach

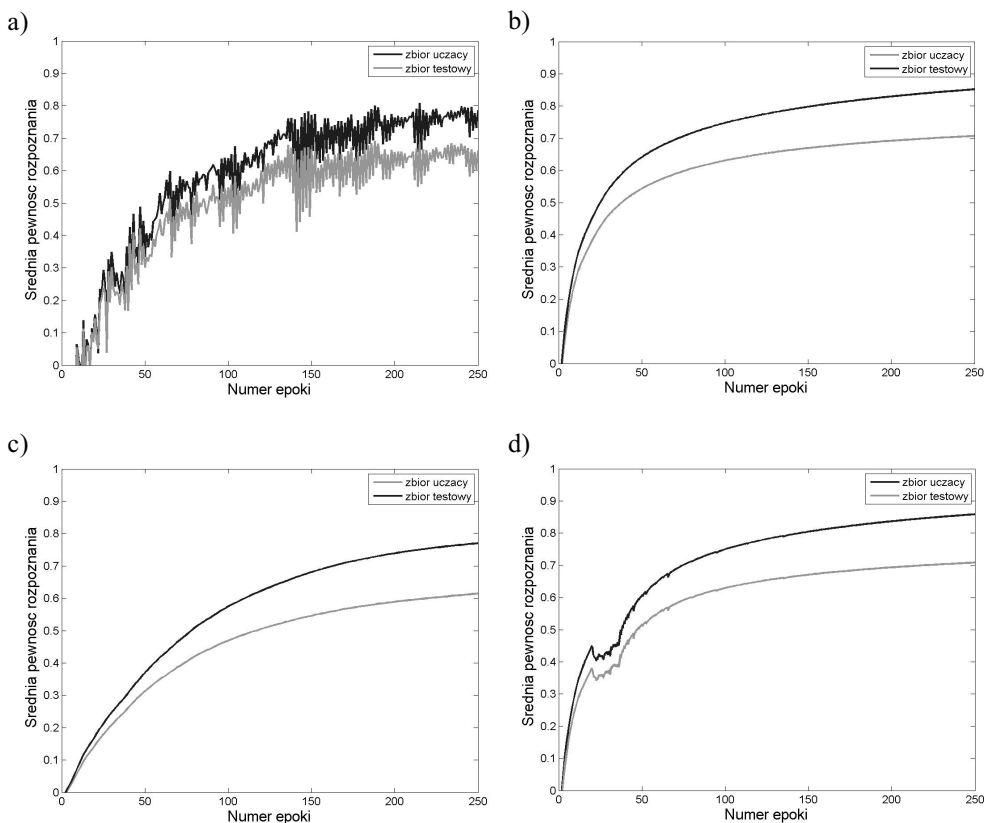
	Uczenie	Algorytm	Średnia pewność rozpoznania	Rozpoznawanie zb. uczącego	Rozpoznawanie zb. testowego
Skuteczność rozpoznania	batchowe	–	0,7639	99,65%	90,07%
	sekwencyjne	losowy	0,8296	98,70%	91,25%
		najgorsze egzemplarze	0,7387	92,00%	85,15%
		matryca pomyłek	0,8280	100,00%	91,30%

Najlepsze wyniki uzyskano w przypadku wykorzystania matrycy pomyłek oraz w uczeniu losowym. Obydwie te metody prezentują zbliżoną jakość rozpoznawania. Metoda z matrycą pomyłek nieco lepiej dostosowuje się do zbioru uczącego, ale skutkuje to nieznacznym osłabieniem pewności rozpoznania.

Opierając się wyłącznie na wynikach w 200 iteracji algorytmu, wolno byłoby postawić hipotezę o całkowitym fiasku metody bazującej na najgorzej rozpoznawanych egzemplarzach. Nie dość, że skuteczność rozpoznania okazała się być fatalna, to jeszcze pewność rozpoznania kształtuje się najgorzej. Okazuje się jednak, że kontynuacja uczenia przy wykorzystaniu tej metody do ok. 500 epoki pozwala na osiągnięcie najwyższej rozpoznawalności zbioru testowego (93,30%), podczas gdy skuteczność rozpoznawania przy użyciu pozostałych metod nie przekracza 91,50%.

Interesujące jest obserwowanie kształtowania się pewności rozpoznawania w trakcie procesu uczenia. Przejście z poziomu czysto losowego na uczenie z matrycą pomyłek spowodowało chwilowe pogorszenie wszystkich wskaźników skuteczności uczenia. Po

stosunkowo krótkim czasie, zarówno pewność jak i skuteczność rozpoznawania przy sieci uczoney matrycą pomyłek okazała się być większa od metody czysto losowej (por. rys. 4d). „Poszarpany” wykres uczenia metodą batchową wynika z zastosowania metody uczenia *resilient backpropagation*.

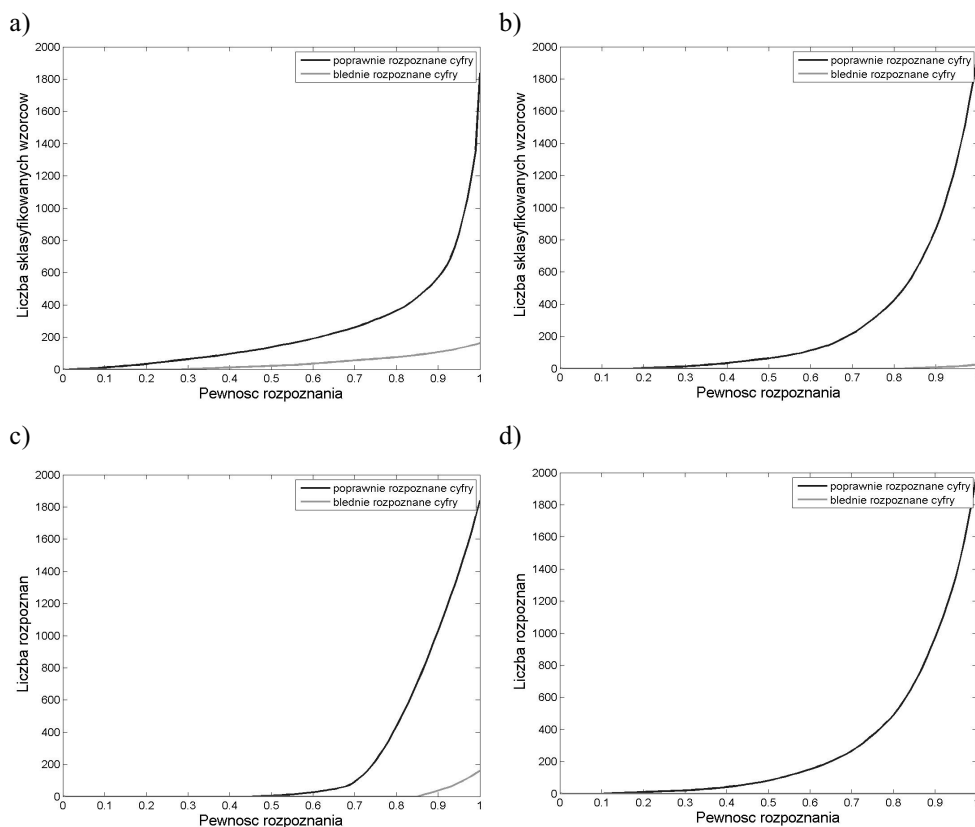


**Rys. 4.** Przebieg pewności rozpoznawania w procesie uczenia sieci neuronowej:  
a) metoda batchowa; b) losowa; c) najgorsze egzemplarze; d) macierz pomyłek

Jeszcze ciekawsze wyniki uzyskuje się analizując skumulowane histogramy pewności rozpoznania dla poszczególnych metod w 200 epoce. (por. rys. 5). Wyniki pokazują, że nie ma sensu stosowanie metody odrzucania przypadków co do których skuteczność klasyfikacji jest wątpliwa, wg algorytmu analogicznego do [9]. Odrzucalibyśmy bowiem znacznie więcej poprawnie rozpoznawanych niż błędnie. Rysunek 5c pokazuje, że metoda najgorszych przypadków pozwala na postawienie znacznie wyższego progu odrzuceń, przy zachowaniu skuteczności rozpoznania. Efektem ubocznym jest większa podatność na błędną klasyfikację wzorców.

Metoda wykorzystująca matrycę pomyłek jako jedyna poprawnie klasyfikowała wszystkie wzorce ze zbioru uczącego (por. rys. 5d).





**Rys. 5.** Skumulowane histogramy pewności poprawnych oraz błędnych rozpoznań dla zbioru uczącego: dla poprawnie i błędnie sklasyfikowanych wzorców: a) metoda batchowa; b) losowa; c) najgorsze egzemplarze; d) macierz pomyłek

## 5. Podsumowanie

W pracy zaproponowano dwie niestandardowe metody uczenia sieci neuronowej, oparte na analizie pewności rozpoznania cyfr ze zbioru uczącego oraz statystyce pomyłek klasyfikacyjnych sieci. W wyniku przeprowadzonych eksperymentów stwierdzono, że stosowanie niestandardowych technik uczenia sieci neuronowej może prowadzić do poprawy skuteczności rozpoznawania wzorców. Dowiedziono również eksperymentalnie, że poprawie skuteczności rozpoznawania wzorców nie musi towarzyszyć zwiększenie pewności rozpoznania.

Uczenie sekwencyjne sieci neuronowej prowadzić może do skuteczniejszego rozpoznawania wzorców od klasycznych metod, niesie jednak za sobą spore obciążenie czasowe. Algorytmy bazujące na pewności rozpoznania w każdej iteracji wymagają niezwykle czasochłonnego sprawdzania całego zbioru uczącego, dają jednak nadzieję na poprawę efektywności rozpoznawania.

Ciekawym tematem do dalszych prac jest kontrola na bieżąco w procesie uczenia pewności rozpoznania poszczególnych wzorców. Po zaprezentowaniu sieci kolejnego, można obserwować zmianę pewności rozpoznawania wszystkich wzorców w zbiorze uczącym – i na tej podstawie usiłować stwierdzić, w jakim stopniu są poszczególne wzorce przyswajalne przez sieć. W przypadku braku poprawy pewności rozpoznawania w przeciągu określonej liczby epok, rozważyć należy czy dalsze forsowanie wzorca ma jakkolwiek sens – i czy nie warto pogodzić się z jego błędną klasyfikacją. Błędnie rozpoznawane przypadki mogą być bowiem na tyle niestandardowe, że ich poprawne rozpoznanie byłoby kłopotliwe także dla nauczyciela. Odrzucenie skrajnych przypadków w procesie uczenia prowadzić więc może do polepszenia zdolności generalizacyjnych.

Warto rozważyć co jest lepsze: kontynuowanie uczenia po odrzuceniu najgorzej przyswajalnych egzemplarzy, czy też wycofanie spowodowanych przezeń zmian – powrót do stanu sprzed prezentacji złego wzorca i uczenie sieci ograniczonym już zbiorem uczącym.

## Literatura

- [1] Behnke S.: *Hierarchical Neural Networks for Image Interpretation*. Computer Science, vol. 2766, Dec. 2003
- [2] Bhattacharya U., Vajda S., Mallick A., Chaudhuri B. B., Belaid A.: *On the Choice of Training Set, Architecture and Combination Rule of Multiple MLP Classifiers for Multiresolution Recognition of Handwritten Characters*. Proceedings of the 9th International Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004)
- [3] Gorgevik D., Cakmakow D.: *An Efficient Three-Stage Classifier for Handwritten Digit Recognition*. Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04), 2004
- [4] Grother P.J.: *Karhunen Loeve Feature Extraction for Neural Handwritten Character Recognition*. Proceedings of Applications of Artificial Neural Networks III, SPIE, Orlando, Florida, 1992, 155–166
- [5] Jain A.K., Duin R.P.W., Mao J.: *Statistical Pattern Recognition: A Review*. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 1, January 2000
- [6] LeCun Y., Jackel L.D., Bottou L., Cortes C., Denker J.S., Drucker H., Guyon I., Muller U.A., Sackinger E., Simard P., Vapnik V.: *Learning Algorithms For Classification: A Comparison On Handwritten Digit Recognition*. [w:] *Neural Networks: The Statistical Mechanics Perspective*, (Oh J. H., Kwon C., Cho S., eds.), 1995, 261–276
- [7] LeCun Y., Bottou L., Orr G., Muller K.: *Efficient BackProp*. [w:] *Neural Networks: Tricks of the trade* (G. Orr and Muller K. eds.), 1998
- [8] Liu X., Rowlands H.: *A step towards Automating the Entering Process of Student Examination Results using Neural Network and Machine Vision Techniques*. 1998
- [9] Mikrut Z., Piaskowska A.: *Attention improves the Recognition Reliability of Backpropagation Network*. Proceedings of 8th International Conference Artificial Intelligence and Soft Computing (ICAISC), 2006
- [10] Tadeusiewicz R.: *Sieci Neuronowe*. Warszawa, Akademicka Oficyna Wydawnicza RM 1993
- [11] Trier C.D., Jain A.K., Taxt T.: *Feature Extraction Methods for Character Recognition – a Survey*. Pattern Recognition, vol. 29, 1996, 641–662