

**Krzysztof BARTECKI**

POLITECHNIKA OPOLSKA, INSTYTUT AUTOMATYKI I INFORMATYKI  
ul. Sosnkowskiego 31, 45-272 Opole

## Klasyczna i neuronowa analiza głównych składowych na przykładzie zadania kompresji obrazu

Dr inż. Krzysztof BARTECKI

Stopień naukowy doktora nauk technicznych uzyskał w 2004 roku w Politechnice Opolskiej. Pracuje w Instytucie Automatyki i Informatyki Politechniki Opolskiej na stanowisku adiunkta. Główne kierunki badań naukowych obejmują: modelowanie i aproksymację układów o parametrach rozłożonych oraz zastosowanie sztucznych sieci neuronowych w zagadnieniach modelowania, identyfikacji oraz sterowania obiektami dynamicznymi.



e-mail: k.bardecki@po.opole.pl

### Streszczenie

W artykule omówiono zastosowanie analizy składników głównych (PCA) w zadaniu kompresji stratnej sygnału na przykładzie kompresji obrazu. Zadanie zrealizowano z wykorzystaniem klasycznej metody PCA oraz dwóch rodzajów sieci neuronowych: jednokierunkowej, dwuwarstwowej sieci z uczeniem nadzorowanym i jednowarstwowej sieci z uczeniem nienadzorowanym. W każdym z przypadków przeanalizowano wpływ struktury modelu PCA na wartości współczynnika kompresji oraz średniokwadratowego błędu kompresji.

**Słowa kluczowe:** analiza składników głównych, sztuczna sieć neuronowa, kompresja danych.

### Classical and neural network-based principal component analysis for image compression

#### Abstract

In the paper, lossy data compression techniques based on the principal component analysis (PCA) are considered on the example of image compression. The presented task is performed using the classical PCA method based on the eigen-decomposition of the image covariance matrix as well as two different kinds of artificial neural networks. The first neural structure used is a two-layer feed-forward network with supervised learning shown in Fig.1, while the second one is a single-layered network with unsupervised Hebbian learning. In each case considered, the effect of the PCA model structure on the data compression ratio and the mean square reconstruction error is analysed. The compression results for a Hebbian neural network with  $K=4$  PCA units are presented in Figs. 2, 3 and 4. They show that only 4 eigenvectors are able to capture the main features of the processed image, giving as a result high value of the data compression ratio. However, the reconstructed image quality is not sufficient from a practical point of view. Therefore, selection of the appropriate value for  $K$  should take into account the tradeoff between a sufficiently high value for the compression ratio and a reasonably low value for the image reconstruction error. The summary results for both classical and neural PCA compression approaches obtained for different number of eigenvectors (neurons) are compared in Fig. 5. The author concludes that a positive aspect of using neural networks as a tool for extracting principal components from the image data is that they do not require calculating the correlation matrix explicitly, as in the case of the classical PCA-based approach.

**Keywords:** principal component analysis, artificial neural network, data compression.

### 1. Wprowadzenie

Analiza głównych składowych (ang. *Principal Component Analysis*, PCA) jest statystycznym narzędziem analizy czynnikowej, w zależności od obszaru zastosowań znanym również pod nazwą metody dekompozycji ortogonalnej, transformaty Hotellinga lub transformaty Karhunen-Loèvego [1, 2, 3]. Typowym obszarem zastosowań PCA jest redukcja wymiarowości danych. Zadanie to polega na opisanu zbioru danych o dużym wymiarze za pomocą

mniejszego zbioru danych, przy jednoczesnym zachowaniu jak największej ilości zawartej w tym zbiorze informacji. Zalety metody PCA wynikają głównie z jej optymalności, sprowadzającej się do maksymalizacji wariancji danych wejściowych oraz minimalizacji średniokwadratowego błędu ich aproksymacji.

Pod względem obliczeniowym PCA bazuje na wyznaczeniu wektorów i wartości własnych macierzy kowariancji danych wejściowych, co przy dużej ich wymiarowości może prowadzić do znacznej złożoności numerycznej. W tym przypadku korzystniejsze może okazać się zastosowanie metod nie wymagających jawnego wyznaczenia macierzy kowariancji, np. wykorzystujących właściwości sztucznych sieci neuronowych (ang. *Artificial Neural Networks*, ANN). Algorytmy neuronowe przetwarzają bezpośrednio dane wejściowe, które mogą być dostarczane zarówno w trybie *off-line* jak i *on-line*. Z tego względu, obok klasycznego podejścia do zadania analizy głównych składowych, w literaturze spotkać można przykłady wykorzystania w tym celu wspomnianych metod neuronowych, np. [4].

W artykule zaprezentowano wyniki zastosowania klasycznej i neuronowej metody PCA w zadaniu kompresji stratnej sygnału na typowym przykładzie kompresji obrazu [5, 6]. Jednak przedstawiona metoda może zostać zastosowana w odniesieniu do dowolnych sygnałów, dla których celowe i możliwe będzie zastosowanie kompresji stratnej, np. w elektroenergetyce [7].

### 2. Klasyczna metoda PCA w zadaniu kompresji obrazu

Zakłada się, że w wyniku operacji dyskretyzacji oraz kwantyzacji obrazu otrzymano zbiór danych  $y(i,j)$  reprezentujących dyskretne wartości luminancji jego pikseli, przy czym  $i=1,2,\dots,M$  oraz  $j=1,2,\dots,N$  są współrzędnymi określającymi miejsce piksela w obrazie, zaś  $M$  i  $N$  reprezentują liczbę punktów próbkowania (rozdzielczość) obrazu, odpowiednio w poziomie i w pionie. Dane te, po odjęciu od każdej próbki  $y(i,j)$  średniej wartości luminancji  $y_m$  dla  $i$ -tego wiersza, wynoszącej

$$y_m(i) = \frac{1}{N} \sum_{j=1}^N y(i,j), \quad (1)$$

reprezentowane są przez macierz  $Y \in \mathbb{R}^{M \times N}$  następującej postaci:

$$Y = [y(i,j) - y_m(i)]. \quad (2)$$

Rozpatrywane zadanie polega na wyznaczeniu macierzy aproksymowanej  $\hat{Y} \in \mathbb{R}^{M \times N}$  takiej, że norma Frobeniusa macierzy błędu aproksymacji  $E = Y - \hat{Y}$ , z definicji równa

$$\|E\|_F = \|Y - \hat{Y}\|_F = \sqrt{\sum_{i=1}^M \sum_{j=1}^N (y(i,j) - \hat{y}(i,j))^2}, \quad (3)$$

osiąga wartość minimalną.

Zastosowanie analizy głównych składowych w powyższym zadaniu sprowadza się do wyznaczenia macierzy  $\hat{Y}$  w oparciu o następującą zależność [1]:

$$\hat{Y} = \Phi_K \Psi_K, \quad (4)$$

gdzie macierz  $\Phi_K \in \mathbb{R}^{M \times K}$  składa się z  $K < M$  pierwszych kolumn ortogonalnej macierzy  $\Phi \in \mathbb{R}^{M \times M}$ , zawierającej  $M$  kolumnowych wektorów

własnych macierzy kowariancji  $C \in \mathbb{R}^{M \times M}$  danych wejściowych, wyznaczanej z zależności:

$$C = \frac{1}{N} Y Y^T. \quad (5)$$

Macierz  $\Phi$  można wyznaczyć jako rozwiązanie następującego zagadnienia własnego:

$$C\Phi = \Phi\Lambda, \quad (6)$$

gdzie  $\Lambda \in \mathbb{R}^{M \times M}$  jest diagonalną macierzą zawierającą rzeczywiste wartości własne symetrycznej macierzy  $C$  (5), uporządkowane w kolejności malejącej:

$$\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_M), \quad \lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_M. \quad (7)$$

Ortogonalną macierz  $\Psi_K \in \mathbb{R}^{K \times N}$  występującą we wzorze (4) można wyznaczyć z następującej zależności [1]:

$$\Psi_K = \Phi_K^T Y, \quad (8)$$

a stąd, na podstawie (4) i (8), otrzymuje się zależność opisującą związek między „oryginalną” a aproksymowaną macierzą  $Y$ :

$$\hat{Y} = \Phi_K \Phi_K^T Y. \quad (9)$$

Omawiana metoda polega zatem na projekcji oryginalnego zbioru danych  $Y$  do podprzestrzeni, rozpiętej na wektorach własnych macierzy kowariancji (5), odpowiadających jej  $K$  największym wartościom własnym. Dla danej wartości  $K$  projekcja ta jest optymalna w sensie minimum błędu średniokwadratowego (ang. *Mean Square Error*, MSE), wynoszącego:

$$\text{MSE} = \frac{1}{M \cdot N} \|E\|_F^2. \quad (10)$$

Rozpatrywany problem aproksymacji danych można traktować jako zadanie kompresji stratnej, dla której współczynnik kompresji  $C_K$  wyznaczany jest z zależności:

$$C_K = \frac{M \cdot N}{M \cdot K + K \cdot N + M}, \quad (11)$$

reprezentującej stosunek liczby elementów oryginalnej macierzy  $Y$  do sumarycznej liczby elementów macierzy  $\Phi_K$ ,  $\Psi_K$  oraz wektora średnich  $y_m$ , wymaganych do jej aproksymacji. Dobór wartości  $K$  polegał będzie zatem na kompromisie między odpowiednio dużą wartością współczynnika kompresji danych (11) a odpowiednio małą wartością błędu kompresji (10).

Innym istotnym wskaźnikiem, wykorzystywanym w celu obiektywnego określenia podobieństwa skompresowanego obrazu do obrazu oryginalnym jest szczytowy stosunek sygnału do szumu PSNR (ang. *Peak Signal-to-Noise Ratio*). Określony jest on następującą zależnością [5, 6]:

$$\text{PSNR} = 10 \cdot \log_{10} \left( \frac{(2^p - 1)^2}{\text{MSE}} \right) \quad [\text{dB}], \quad (12)$$

gdzie  $p$  jest liczbą bitów przyjętych do zakodowania wartości luminancji pikseli w obrazie (np.  $p=8$  dla kodowania jednobajtowego). Typowe wartości PSNR w przypadku stratnej kompresji obrazu i wideo wynoszą od 30 do 50 dB, przy czym wyższa wartość oznacza lepszą jakość obrazu. Dopuszczalne wartości w przypadku transmisji bezprzewodowej wynoszą około 20-25 dB.

### 3. Neuronowe algorytmy PCA

W niniejszym rozdziale przedstawiono dwie struktury sieci neuronowych realizujących zadanie analizy głównych składowych,

które wykorzystane zostały w rozpatrywanym zadaniu kompresji stratnej sygnału.

#### 3.1. Dwuwarstwowa sieć jednokierunkowa z uczeniem nadzorowanym (FF-PCA)

W rozpatrywanym przypadku wykorzystuje się jednokierunkową, dwuwarstwową sieć z neuronami liniowymi o strukturze przedstawionej na rys. 1. Liczba wyjść sieci (tzn. liczba neuronów jej drugiej warstwy, nazywanej dalej *warstwą dekompresji*) równa jest zarazem liczbie jej wejść i odpowiada liczbie wierszy  $M$  macierzy danych wejściowych  $Y$  (2). Z kolei liczba neuronów pierwszej warstwy sieci  $K$ , zwanej dalej *warstwą kompresji*, odpowiada liczbie wektorów własnych w przypadku klasycznej metody PCA. Omawiana tutaj sieć w dalszej części artykułu oznaczana będzie skrótem FF-PCA (ang. *Feed-Forward Principal Component Analysis*).

Oznaczając przez  $W^{(1)} \in \mathbb{R}^{K \times M}$  macierz współczynników wagowych warstwy kompresji, przez  $W^{(2)} \in \mathbb{R}^{M \times K}$  macierz współczynników wagowych warstwy dekompresji, zaś przez  $V \in \mathbb{R}^{K \times N}$  – macierz odpowiedzi warstwy kompresji na wzorce wejściowe  $Y$ , otrzymuje się następujące zależności opisujące działanie sieci w postaci wsadowej, tzn. jednocześnie dla wszystkich wzorców wejściowych składających się na macierz  $Y$ :

$$V = W^{(1)} Y, \quad (13)$$

oraz

$$\hat{Y} = W^{(2)} V = W^{(2)} W^{(1)} Y. \quad (14)$$

Wzór (14) stanowi odpowiednik zależności (4) oraz (9) dla klasycznej metody PCA, przy czym macierz wagowa  $W^{(1)}$  odpowiada transponowanej macierzy  $\Phi_K$ , zaś macierz odpowiedzi pierwszej warstwy  $V$  – macierzy  $\Psi_K$ . Rezultat uzyskany w wyniku minimalizacji funkcji (10) przy użyciu procedury uczenia rozpatrywanej sieci neuronowej jest jednak niejednoznaczny w odniesieniu do rozwiązania dostarczanego przez opisaną w poprzednim punkcie klasyczną metodą PCA, gdyż każda macierz będąca odpowiednią liniową transformacją macierzy  $W^{(1)}$  i  $W^{(2)}$  spełniać będzie rozpatrywane zadanie autoasocjacji [8].

Rolę wejściowych wzorców uczących sieci pełnią kolejne kolumny macierzy  $Y$  (2). Uczenie ma charakter autoasocjacyjny, tzn. wzorce wyjściowe równe są odpowiednim wzorcom wejściowym, zaś procedura uczenia polega na przeprowadzanych iteracyjnie modyfikacjach wartości współczynników wagowych sieci w celu minimalizacji jej funkcji błędu, stanowiącej bezpośredni odpowiednik funkcji błędu aproksymacji macierzy  $Y$  (10). Najczęściej wykorzystuje się tu gradientową metodę największego spadku z adaptacyjnymi zmianami współczynnika prędkości uczenia oraz z tzw. członem *momentum*, w oparciu o następującą zależność [1, 8, 9]:

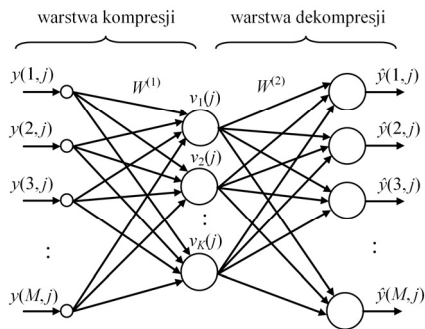
$$w(t+1) = w(t) - \eta(t) \nabla E(w(t)) + \alpha(t) (w(t) - w(t-1)), \quad (15)$$

gdzie:  $w(t)$  – wektor wag sieci w kroku uczenia  $t$ ,  $\nabla E(w(t))$  – gradient funkcji błędu sieci względem jej wag,  $\eta(t)$  – współczynnik prędkości uczenia,  $\alpha(t)$  – stała *momentum*.

Znacznie korzystniejszy przebieg procedury uczenia sieci uzyskuje się stosując quasi-newtonowskie metody optymalizacyjne, a w szczególności algorytm Levenberga-Marquardta. Modyfikacja wartości wag sieci realizowana jest tu według następującego wzoru [1, 8, 9]:

$$w(t+1) = w(t) - \eta(t) [\tilde{H}(w(t))]^{-1} \nabla E(w(t)), \quad (16)$$

gdzie  $\tilde{H}(w(t))$  jest aproksymowaną macierzą *hesjanu*, czyli macierzą zawierającą drugie pochodne funkcji błędu sieci (10) względem jej wszystkich współczynników wagowych.



Rys. 1. Struktura jednokierunkowej, dwuwarstwowej sieci neuronowej PCA  
Fig. 1. Structure of the two-layer feed-forward PCA neural network

Obydwa przedstawione tu algorytmy uczenia możliwe są do implementacji zarówno w wersji *wsadowej* (ang. *batch training*), w przypadku której modyfikacja wag sieci odbywa się po prezentacji całego zbioru danych uczących, jak również w wersji adaptacyjnej, w której wagi modyfikowane są każdorazowo po prezentacji pojedynczego wzorca (wektora) uczącego. To drugie podejście umożliwia wykorzystanie sieci neuronowej do aproksymacji macierzy  $Y$  w trybie *on-line*, wraz z napływającymi danymi, tworzącymi kolejne jej kolumny.

### 3.2. Sieć jednowarstwowa z uczeniem nienadzorowanym (GHA-PCA)

Alternatywne podejście do ekstrakcji składników głównych z zbioru danych wejściowych może polegać na zastosowaniu sieci neuronowej z uczeniem nienadzorowanym, np. wykorzystującej tzw. uogólniony algorytm Hebba (ang. *Generalized Hebbian Algorithm*, GHA). Metoda ta w odniesieniu do pojedynczego neuronu po raz pierwszy zaproponowana została w pracach Oji [10], zaś jej rozszerzenie na sieć złożoną z wielu neuronów, umożliwiającą estymację kolejnych składników głównych, zaprezentowano m. in. w pozycjach Oji i Sangera [11, 12]. W tym przypadku zadanie realizowane jest z zastosowaniem jednowarstwowej sieci neuronowej, złożonej z  $K$  neuronów liniowych, odpowiadającym kolejnym głównym składnikom. Struktura wykorzystanej w rozpatrywanym zadaniu sieci GHA-PCA odpowiada warstwie kompresji sieci FF-PCA z rys. 1. Jednak w odróżnieniu od niej, w procedurze uczenia sieci GHA-PCA nie wykorzystuje się, zgodnie z metodologią uczenia nienadzorowanego, wzorców wyjściowych, lecz jedynie wzorce wejściowe oraz wartości pojawiające się na wyjściach sieci.

Przyjmując oznaczenia jak na rys. 1, oraz zakładając że uczenie sieci przebiega w trybie *on-line*, tzn. modyfikacja jej współczynników wagowych odbywa się każdorazowo po prezentacji wzorca wejściowego odpowiadającego kolejnej kolumnie macierzy obrazu  $Y(1)$ ,  $k$ -ty neuron sieci generuje sygnał wyjściowy zgodnie z następującą zależnością:

$$v_k(j) = \sum_{i=1}^M w_{k,i}(j) y(i, j), \quad (17)$$

gdzie  $w_{k,i}(j)$  oznacza wartość współczynnika wagowego wiążącego  $k$ -ty neuron z  $i$ -tym wejściem sieci, wyznaczoną dla wzorca zawartego w  $j$ -tej kolumnie macierzy  $Y$ .

W procedurze uczenia omawianej sieci zastosowano uogólniony algorytm Hebba, znany również pod nazwą reguły Sangera, stanowiący uogólnienie wspomnianej wcześniej reguły Oji na sieć złożoną z wielu neuronów. Modyfikacja wartości poszczególnych współczynników wagowych realizowana jest tu w oparciu o następującą zależność [8, 12]:

$$w_{k,i}(j) = w_{k,i}(j) + \eta v_k(j) \left[ y(i, j) - \sum_{h=1}^k w_{h,i}(j) v_h(j) \right] \quad (18)$$

dla  $i = 1, 2, \dots, M$  oraz  $k = 1, 2, \dots, K$ .

Oznaczając przez  $w_k(j)$  wektor wszystkich wag  $k$ -tego neuronu, wyznaczonych dla  $j$ -tego wzorca wejściowego, tzn. wektor następującej postaci:

$$w_k(j) = [w_{k,1}(j) \ w_{k,2}(j) \ \dots \ w_{k,M}(j)], \quad (19)$$

przez  $y(j)$  wektor wejściowy sieci reprezentujący  $j$ -tą kolumnę macierzy obrazu  $Y(2)$ :

$$y(j) = [y(1, j) \ y(2, j) \ \dots \ y(M, j)]^T, \quad (20)$$

oraz wprowadzając następujące wyrażenie:

$$y'(j) = y(j) - \sum_{h=1}^{k-1} (w_h(j))^T v_h(j), \quad (21)$$

zależność (18) można zapisać w postaci wektorowej:

$$w_k(j+1) = w_k(j) + \eta v_k(j) \left[ (y'(j))^T - w_k(j) v_k(j) \right], \quad (22)$$

analogicznej do algorytmu Oji obowiązującego dla pojedynczego neuronu, dla którego zachodzi samonormalizacja wektorów współczynników wagowych.

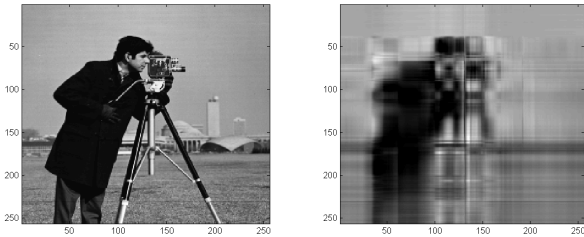
Jak wspomniano na wstępie artykułu, jednym z głównych zastosowań transformacji PCA jest kompresja danych. W rozpatrywanym przypadku zadanie kompresji należy rozumieć następująco: duża liczba danych wejściowych, reprezentujących rozkład luminancji poszczególnych pikseli obrazu jest zastępowana zmniejszoną liczbą danych, zawartą we współczynnikach wagowych oraz odpowiedziach sieci. W celu odtworzenia oryginalnego obrazu wymagana jest dekompresja, przeprowadzana w oparciu o macierz wag i macierz odpowiedzi sieci, analogicznie jak w przypadku zależności (4) oraz (14).

### 4. Wyniki kompresji obrazu z zastosowaniem metod PCA

Omówione w poprzednich punktach algorytmy kompresji PCA zaimplementowane zostały w środowisku do obliczeń numerycznych *Matlab*. Jako obraz testowy wykorzystano dołączony do programu plik *cameraman.tif* o rozdzielczości 256×256 pikseli, zakodowany w 8-bitowej skali szarości. Na rys. 2–4 przedstawiono wyniki uzyskane dla kompresji z zastosowaniem sieci GHA-PCA złożonej z czterech neuronów, tzn. przy uwzględnieniu czterech wektorów własnych macierzy kowariancji obrazu. Rys. 2 przedstawia obraz oryginalny (po lewej) oraz odtworzony (po prawej). Ze względu na małą liczbę wektorów własnych, błąd kompresji (10) ma dużą wartość i jakość obrazu po rekonstrukcji nie jest zadowalająca. Rys. 3 przedstawia graficzną ilustrację wektorów współczynników wagowych poszczególnych neuronów nauczonej sieci. Wartości tych współczynników reprezentują wektory własne macierzy kowariancji  $C(5)$ . Z kolei na rys. 4 przedstawiono przestrzenny wykres błędu kompresji  $e(i, j) = y(i, j) - \hat{y}(i, j)$ . Lepszą jakość zdekompresowanego obrazu, oczywiście kosztem mniejszej wartości współczynnika kompresji, uzyskano zwiększając liczbę neuronów  $K$ .

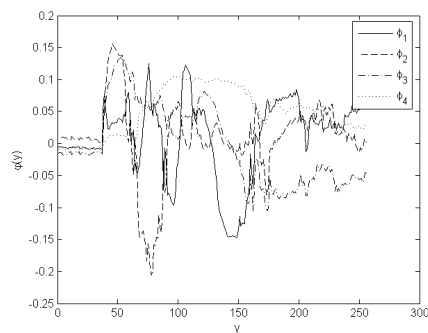
Analogiczne badania przeprowadzono wykorzystując również klasyczną metodę PCA oraz sieć neuronową FF-PCA. Kompresję przeprowadzono dla ośmiu różnych wartości parametru  $K$ , wynoszących odpowiednio: 1, 2, 4, 8, 16, 32, 64 oraz 128. Jej wyniki przedstawiono na rys. 5, na którym dla każdego z wymienionych przypadków podano uzyskane wartości współczynnika kompresji  $C_K(11)$  oraz szczytowego stosunku sygnału do szumu PSNR (12). Jak wynika z otrzymanych rezultatów, wraz ze wzrostem wartości parametru  $K$  rośnie wartość współczynnika PSNR reprezentującego tłumienie występujących w sygnale obrazu zniekształceń spowodowanych kompresją. Jednak jednocześnie maleje wówczas wartość współczynnika kompresji  $C_K$ . Z tego względu dobór odpowiedniej wartości parametru  $K$  powinien uwzględniać kompromis między dostatecznie małą wartością średniokwadratowego

błędu kompresji, oraz dostatecznie dużą wartością współczynnika kompresji.



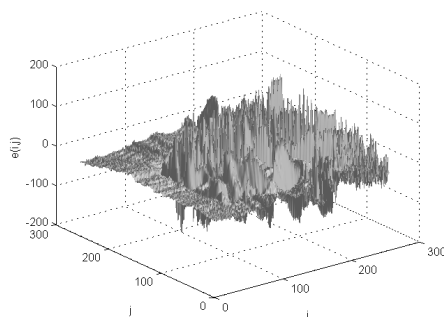
Rys. 2. Kompresja z zastosowaniem sieci neuronowej GHA-PCA dla  $K=4$ :

po lewej obraz oryginalny, po prawej – zrekonstruowany  
Fig. 2. Image compression results for the GHA-PCA neural network ( $K=4$ ): original (left) and reconstructed (right) images



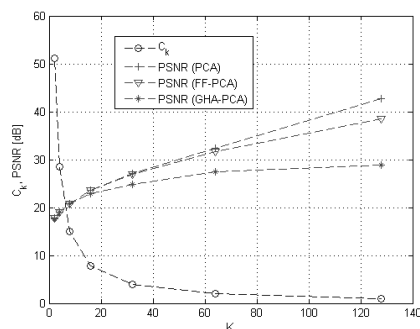
Rys. 3. Wektory współczynników wagowych nauczonej sieci GHA-PCA

reprezentujące wektory własne macierzy kowariancji obrazu dla  $K=4$   
Fig. 3. Weight vectors of the trained GHA-PCA network representing the eigenvectors of the image covariance matrix for  $K=4$



Rys. 4. Wykres błędu kompresji  $e(i,j)$  obrazu dla kompresji neuronową metodą GHA-PCA ( $K=4$ )

Fig. 4. Image compression error  $e(i,j)$  for the neural GHA-PCA compression method ( $K=4$ )



Rys. 5. Zależność parametrów  $C_k$  oraz PSNR od liczby wektorów własnych (neuronów)  $K$  dla rozpatrywanych metod kompresji PCA

Fig. 5. The dependence of parameters  $C_k$  and PSNR on the number of eigenvectors (neurons)  $K$  for the considered PCA compression methods

Z kolei porównując wyniki uzyskane dla poszczególnych metod można stwierdzić, iż najlepsze wyniki uzyskano dla kompresji przy pomocy klasycznej metody PCA. Zbliżone wyniki uzyskano dla jednokierunkowej sieci neuronowej z uczeniem nadzorowanym (FF-PCA). Najslabiej wypada tu metoda wykorzystująca sieć jednowarstwową z adaptacyjnym uczeniem nienadzorowanym zgodnie z uogólnioną regułą Hebba (GHA-PCA). Różnice szczególnie widoczne są dla dużych wartości  $K$ , tzn. dla małych wartości współczynnika  $C_k$ .

## 5. Podsumowanie

W artykule zaprezentowano możliwość wykorzystania analizy głównych składowych (PCA) w zadaniu stratnej kompresji obrazu. Zadanie zrealizowano stosując zarówno podejście klasyczne, bazujące na wyznaczeniu wektorów i wartości własnych macierzy kowariancji danych, jak również wykorzystując dwa różne rodzaje sztucznych sieci neuronowych: jednokierunkową, dwuwarstwową sieć z uczeniem nadzorowanym (FF-PCA) i jednowarstwową sieć z uczeniem nienadzorowanym (GHA-PCA).

Przeanalizowano wpływ liczby  $K$  wektorów własnych na wartości współczynnika kompresji oraz średniokwadratowego błędu kompresji. Zastosowanie w zadaniu estymacji głównych składowych sieci neuronowej z uczeniem nadzorowanym (FF-PCA) dało pod względem uzyskanych wartości błędu kompresji rezultaty zbliżone do otrzymanych przy użyciu klasycznej metody PCA. Jedynym mankamentem algorytmu neuronowego był nieco dłuższy czas wymagany do wyznaczenia odpowiednich wartości parametrów modelu (współczynników wagowych), wynikający z zastosowanej iteracyjnej procedury uczenia sieci. Z kolei wyniki uzyskane dla metody GHA-PCA były gorsze od wyników otrzymanych dla dwóch wcześniej wymienionych metod, w szczególności w przypadku małej wartości współczynnika kompresji. Korzystnym aspektem zastosowania neuronowych metod PCA jest natomiast fakt, iż można je stosować w przypadku akwizycji danych pomiarowych w trybie *on-line*, kiedy to utworzenie jawnej postaci macierzy korelacji danych wejściowych jest niemożliwe.

## 6. Literatura

- [1] Bartecki K.: Neural network-based PCA: An application to approximation of a distributed parameter system. Lecture Notes in Computer Science, vol. 7267 (2012), 3-11.
- [2] Berkooz G., Holmes P., Lumley J.L.: The proper orthogonal decomposition in the analysis of turbulent flows. Annual Review of Fluid Mechanics, vol. 25 (1993), 539-575.
- [3] Chatterjee A.: An introduction to the proper orthogonal decomposition. Current Science, vol. 78, no. 7 (2000), 808-817.
- [4] Diamantaras K. I., Kung S. Y.: Principal Component Neural Networks – Theory and Applications. John Wiley, New York (1996).
- [5] Cierniak R., Rutkowski L.: Nowe metody kompresji obrazu. Wydawnictwo Gnome, Gliwice (2000).
- [6] Skarbek W. (red.): Multimedia – Algorytmy i Standardy Kompresji. Akademicka Oficyna Wydawnicza, Warszawa (1998).
- [7] Zygarlicki J., Mroczka J.: Sposób doboru metod kompresji dla sygnałów z sieci elektroenergetycznej. Pomiary Automatyka Kontrola, nr 10 (2006), 35-38.
- [8] Osowski S.: Sieci neuronowe do przetwarzania informacji. OW Politechniki Warszawskiej (2000).
- [9] Bartecki K., Czorny M.: Implementacja sztucznej sieci neuronowej w architekturze równoległej z wykorzystaniem protokołu MPI. Pomiary Automatyka Kontrola, vol. 57, nr 6, (2011), 638-640.
- [10] Oja E.: A simplified neuron model as a principal component analyzer. Journal of Mathematical Biology, vol. 15, no 3, (1982), 267-273.
- [11] Oja E.: Neural networks, principal components and subspaces. International Journal of Neural Systems, vol. 1, no 1 (1989), 61-68.
- [12] Sanger T. D.: Optimal unsupervised learning in a single-layer linear feedforward neural network. Neural Networks, vol. 2 (1989), 459-473.