

Article

# Inpainted Image Reconstruction Using an Extended Hopfield Neural Network Based Machine Learning System

Wieslaw Citko \*  and Wieslaw Sienko

Department of Electrical Engineering, Gdynia Maritime University, Morska 81-87, 81-225 Gdynia, Poland; w.sienko@we.umg.edu.pl

\* Correspondence: w.citko@we.umg.edu.pl

**Abstract:** This paper considers the use of a machine learning system for the reconstruction and recognition of distorted or damaged patterns, in particular, images of faces partially covered with masks. The most up-to-date image reconstruction structures are based on constrained optimization algorithms and suitable regularizers. In contrast with the above-mentioned image processing methods, the machine learning system presented in this paper employs the superposition of system vectors setting up asymptotic centers of attraction. The structure of the system is implemented using Hopfield-type neural network-based biorthogonal transformations. The reconstruction property gives rise to a superposition processor and reversible computations. Moreover, this paper's distorted image reconstruction sets up associative memories where images stored in memory are retrieved by distorted/inpainted key images.

**Keywords:** artificial intelligence; machine learning; image reconstruction and recognition



**Citation:** Citko, W.; Sienko, W. Inpainted Image Reconstruction Using an Extended Hopfield Neural Network Based Machine Learning System. *Sensors* **2022**, *22*, 813. <https://doi.org/10.3390/s22030813>

Received: 26 November 2021

Accepted: 17 January 2022

Published: 21 January 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Machine learning, a sub-field of artificial intelligence, deals with algorithms that build mathematical models to automatically make decisions or predictions based on sample data called training sets. The concept of learning is the key to understanding intelligence in both biological brain structures and machines. The aim of machine learning is to create mappings  $y = F(x)$ ,  $y \in R^m, x \in R^n$ , generated by training sets  $S = \{x_i, y_i\}_{i=1}^N$ , the vectors of which are approximation nodes. Hence, the training points include:

$$y_i = F(x_i), i = 1, \dots, N \quad (1)$$

The machine learning model described in a previous paper [1] was derived from an extended Hopfield neural network and is based on spectral analysis that uses biorthogonal and orthogonal transformations. It should be emphasized that this system has a universal character that enables the implementation of basic functions of learning systems, such as pattern association, pattern recognition, and inverse modeling. One of the aforementioned properties of this model is the recognition and reconstruction of image patterns. In [1], we presented an example where the object of the reconstruction was an incomplete, inpainted image of a subject named Lena. Such examples of reconstruction allow for the development of a system based on the above-mentioned model of machine learning that can recognize people wearing masks. It is worth noting that the above-mentioned model of machine learning represents an alternative to classical image reconstruction/restoration systems, which make use of such processing tools as inverse modelling, deconvolution, Wiener filters, and PCA (Principal Component Analysis) [2–5].

Classical image reconstruction systems are currently being intensively supplemented and replaced by those using neural, neuro-fuzzy architecture, and algorithms, especially in medical applications [6–10]. A comprehensive review of recent advances in image reconstruction can be found in [11]. Current research is focused on sparsity, low-rankness, and

machine learning [12,13]. It is worth noting that the most up-to-date image reconstruction structures are based on constrained optimization algorithms and adequate regularizers [14].

Recently, deep learning algorithms are driving a renaissance of interest in neural network research and applications (e.g., image processing). Most of the known deep learning algorithms are implemented in the form of ANN (DLNN) learned from training set data by minimizing loss functions. Thus, deep learning approach can be seen as a special topic in optimization theory. Standard types of deep learning neural networks include the multilayer perceptrons (MLP), convolutional neural networks (CNN), recurrent neural networks (RNN), and generative adversarial networks (GAN) [15–18]. However, optimal networks topology and implementation technology have not yet been selected (the generalizability of networks is not well understood, and there is a lack of explanation for the relationship between the network topology and performance [19]). Nevertheless, we claim that ANN should constitute both a universal algorithmic and physical models used in computational intelligence. It is clear that Hopfield-type neural networks are both physical and algorithmic models suitable for neural computations. Hence, we considered an extended Hopfield-type model of the neural network defined by the following equations:

$$\dot{x} = (\eta W - w_0 \mathbf{1} + \varepsilon W_s) \theta(x) + I_d \quad (2)$$

where

- $W$ —skew-symmetric orthogonal matrix;
- $W_s$ —real symmetric matrix;
- $\mathbf{1}$ —identity matrix;
- $\theta(x)$ —vector of activation functions;
- $I_d$ —input vector; and
- $\varepsilon, w_0, \eta$ —parameters.

$$(\eta W - w_0 \mathbf{1} + \varepsilon W_s) \theta(x) + I_d = \mathbf{0} \quad (3)$$

An equilibrium equation of neural networks (2), i.e., gives rise to the universal models of machine learning based on biorthogonal transformations, enabling the realization of common learning systems functions. One of these functions is the implementation of associative memories. Thus, this paper's inpainted image reconstruction system sets up associative memories where images stored in memory are retrieved by distorted/inpainted key images. To summarize, we propose a machine learning model that uses biorthogonal transformations based on spectral processing as alternative solutions to deep learning based on optimization procedures.

The rest of this paper is structured as follows: Section 2 provides details on the proposed learning algorithm and presents a structure of the machine learning systems for image processing. Section 2 contains also some results of computational verifications using MATLAB software. Section 3 includes some results of image processing as an inverse problem. Some unique properties of this machine learning system are discussed in Section 4. The conclusions underline the main features of the machine learning system presented in the article.

## 2. Materials and Methods

### 2.1. Machine Learning System for Image Processing

We consider a set of  $N$  black and white images represented by  $m$  rows and  $n$  columns, i.e., a set of  $(m \cdot n)$  pixels with different shades of grayness. For vector analysis, each image is transformed by concatenating  $m$  rows to form the column vector  $x_i$  ( $m \cdot n \times 1$ ),  $i = 1, \dots, N$ . Thus, the set of  $N$  images is represented by the following matrix:

$$X = [x_1, x_2, \dots, x_N], \dim x_i = m \cdot n = 2^k, k = 3, 4, \dots, \quad (4)$$

where

$$N < \frac{1}{2}(n \cdot m). \quad (5)$$

The set of distorted images is given by the matrix:

$$\mathbf{X}^{(s)} = [\mathbf{x}_1^{(s)}, \mathbf{x}_2^{(s)}, \dots, \mathbf{x}_N^{(s)}]. \quad (6)$$

It is straightforward to observe that the training set is as follows:

$$S = \left\{ \mathbf{x}_i, \mathbf{x}_i^{(s)} \right\}_{i=1}^N. \quad (7)$$

$S$  creates a mapping  $F(\cdot)$  defined by the following properties:

$$\mathbf{x}_i = F(\mathbf{x}_i^{(s)}) \quad (8)$$

and

$$\mathbf{x}_i^{(s)} \xrightarrow{F} \mathbf{x}_i, \quad i = 1, 2, \dots, N. \quad (9)$$

Thus, the mapping  $F$  is implemented as a machine learning system for image reconstruction.

The structure implementing the mapping  $F(\cdot)$  defined by Equations (8) and (9) can be obtained as the solutions of the equilibrium Equation (3). Thus, for  $w_0 = 2$ ,  $\varepsilon = 1$  in Equation (3), one gets:

$$(\mathbf{W}_{2^k} - 2 \cdot \mathbf{1} + \mathbf{W}_s) \mathbf{m}_i + \mathbf{x}_i^{(s)} = \mathbf{0} \quad (10)$$

where  $\mathbf{W}_{2^k}^2 = -\mathbf{1}$ , skew-symmetric, orthogonal matrix

Hence, the  $N$ -solutions are as follows:

$$\mathbf{m}_i = (2 \cdot \mathbf{1} - \mathbf{W}_s - \mathbf{W}_{2^k})^{-1} \mathbf{x}_i^{(s)}, \quad i = 1, \dots, N \quad (11)$$

where

$$\mathbf{W}_s = \mathbf{M}(\mathbf{M}^T \mathbf{M})^{-1} \mathbf{M}^T \quad (12)$$

and

$$\mathbf{M} = \{\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_N\} \quad (13)$$

is a spectrum matrix of given vectors  $\mathbf{x}_i$ , i.e.,

$$\mathbf{m}_i = \frac{1}{2}(\mathbf{W}_{2^k} + \mathbf{1})\mathbf{x}_i \quad (14)$$

and

$$\mathbf{x}_i = (-\mathbf{W}_{2^k} + \mathbf{1})\mathbf{m}_i, \quad i = 1, \dots, N. \quad (15)$$

Equation (11) can be seen as a determination of biorthogonal transformation  $T_s(\cdot)$ :

$$\mathbf{m}_i = T(\mathbf{x}_i^{(s)}) \quad (16)$$

and Equation (14) can be seen as an orthogonal transformation:

$$\mathbf{m}_i = T(\mathbf{x}_i), \quad \mathbf{x}_i = T^{-1}(\mathbf{m}_i). \quad (17)$$

The transformations  $T_s(\cdot)$  and  $T^{-1}(\cdot)$ , arranged as a realization of the mapping  $F(\cdot)$ , have the block structure, as shown in Figure 1 [1]. The orthogonal transformation  $T(\cdot)$ , which makes use of the Hurwitz-Radon matrix family [20], allows for determining the Haar-Fourier spectra of the system vectors  $\mathbf{x}_i$ .

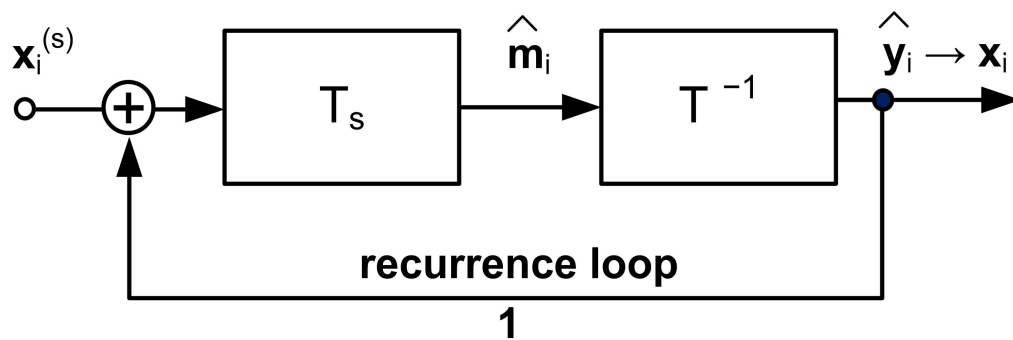


Figure 1. Structure of the machine learning model for image processing.

The structure from Figure 1 serves as the estimator of the spectrum  $\{\hat{m}_i\}$ :

$$\hat{m}_i = T_s(x_i^{(s)}), i = 1, \dots, N. \tag{18}$$

In the system, due to the iterative nature of the feedback loop, the following convergence of vectors is obtained:

$$\hat{m}_i \rightarrow m_i \tag{19}$$

$$\hat{y}_i \rightarrow x_i, i = 1, \dots, N. \tag{20}$$

The convergence determined by Equation (20) is performed in  $K$  iterations ( $K$  depends on the reconstruction problem, note the example shown below). Moreover, it should be noted that for input image  $z \neq x_i, i = 1, \dots, N$ , the output of the system is given by the superposition of system vectors:

$$F(z) = \sum_{i=1}^N \alpha_i x_i, \alpha_i \in R. \tag{21}$$

The system vectors  $x_i$  set up the attraction centers.

The structure in Figure 1 can also be represented as the lumped memory model in Figure 2. It is worth noting that this structure gives rise to the realization of an AI analog processor. However, this topic is beyond the scope of this paper. The synthesis algorithm of the system given in Figure 1 can be found in Appendix A

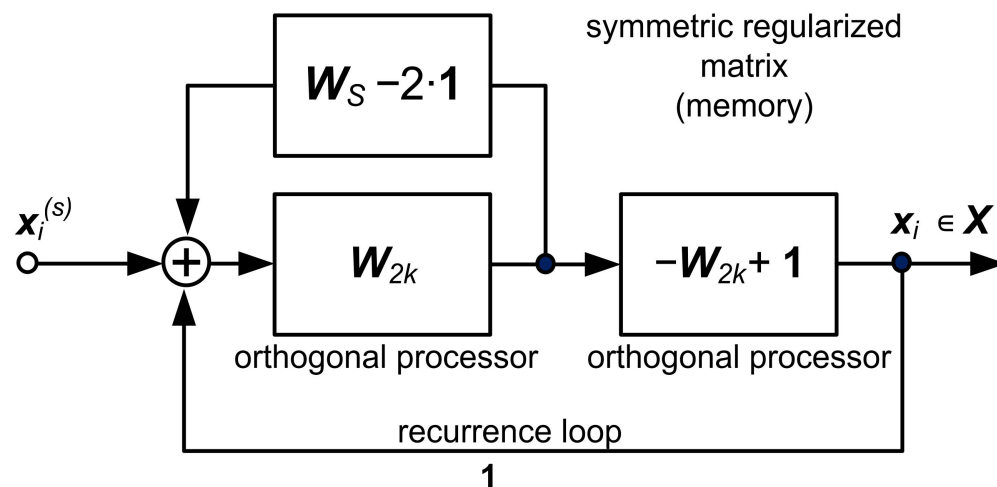


Figure 2. Block diagram of the approximator with lumped memory.

## 2.2. Computational Verification of the Learning Algorithm—Examples of Face Image Reconstruction and Person Recognition

A. The machine image processing system described in the previous section was used to reconstruct and classify a set of images. The system task was to reconstruct a complete face image based on a masked photo (mask applied by software) and to assign the reconstructed image to a specific person. In the system, photos of 9 faces ( $N = 9$ ) were stored in memory in the form of a  $64 \times 64$  matrix defining the degree of grayness of individual image pixels. The saved face images are presented in Figure 3. For vector analysis, each image was transformed by concatenating 64 lines into the form of a column vector  $x_i$  ( $64 \cdot 64 \times 1$ ),  $i = 1, \dots, 9$ . After transformation, the set of 9 images was represented by the matrix  $X$  ( $4096 \times 9$ ):  $X = [x_1, x_2, \dots, x_9]$ ,  $\dim x_i = 64 \cdot 64 = 4096 = 2^k$ ,  $k = 12$ .



Figure 3. Face images saved (source: <https://pixabay.com/pl>, accessed on 17 February 2021).

In the experiments, the identification numbers 1, 2,  $\dots$ , 9 were assigned to the images. The system vectors  $u_i = \begin{bmatrix} x_i \\ i \end{bmatrix}$ ,  $i = 1, 2, \dots, 9$  were used to construct the machine learning system according to the procedure described in the previous section. Examples of the reconstruction of photos of people wearing masks are shown in Figure 4.

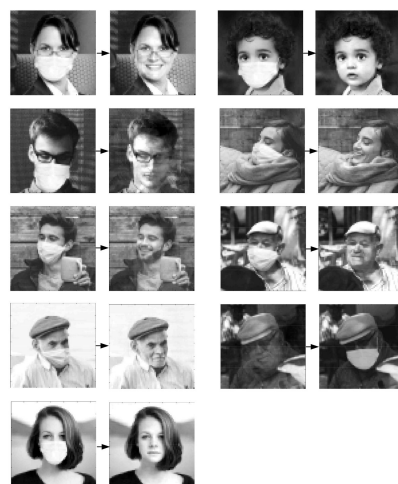


Figure 4. Reconstruction of face images of people wearing masks.

Table 1 shows the nominal values of the Recognition Index, i.e., the assigned numbers and their associated values after 100 iterations. The results presented in Table 1 show that in most cases, the value of the index rounded to the nearest integer corresponds to the nominal value. Thus, the system correctly identifies each person with the exception of Photo Number 3, where the person is incorrectly recognized. Increasing the number of iterations did not change the index, as the process quickly converges to the final value.

**Table 1.** Values of the Recognition Index of each person.

Photo Number	Index Nominal Value	Index Value after 100 Iterations
1	1.0	0.8622
2	2.0	1.6240
3	3.0	2.3660
4	4.0	3.9983
5	5.0	5.1259
6	6.0	5.8842
7	7.0	6.7262
8	8.0	8.0466
9	9.0	8.9576

The convergence of the iterative process is illustrated in Table 2, which presents the index values obtained after successive iterations. The experiment was carried out for Photo Number 2 with the nominal value of the coefficient of 2.0.

**Table 2.** Convergence of the iterative process for Photo Number 2.

Number of Iterations	Index Value	Number of Iterations	Index Value
1	−0.0813	7	1.4607
2	0.1758	8	1.5394
3	0.5332	9	1.5843
4	0.8703	10	1.6078
5	1.1394	12	1.6233
6	1.3327	100	1.6240

A significant result that confirms the principle of the proposed system was obtained by substituting in a photo that was not saved in the system. The response shown in Figure 5 is a superposition of the photos stored in memory in the system (Equation (21)).



**Figure 5.** Attempt to recognize an unsaved photo.

The mean squared error (MSE) values calculation for the previously performed image reconstructions are presented in Table 3. The second column of the table shows the MSE describing the difference between the original and masked photos, whereas the third column shows the error describing the difference between the original photo and the photo after reconstruction. Each time an image saved in the system was analyzed, the mean squared error decreased. For the reconstruction attempt shown in Figure 5, which uses an image not saved in the system, the MSE error is 2950.90.

**Table 3.** Values of the Recognition Index of each person.

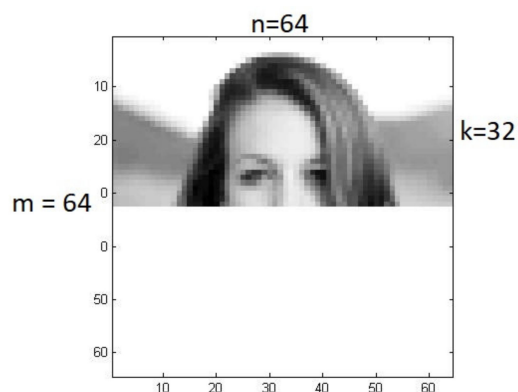
Photo Number	MSE (Original Photo—Ask Photo)	MSE (Original Photo—Reconstructed Photo)
1	366.49	105.06
2	595.96	176.38
3	1573.00	570.95
4	398.00	37.58
5	552.04	114.55
6	675.67	112.13
7	828.53	221.09
8	171.52	26.05
9	327.06	40.75

The fractional value of the index in Table 1 reflects the system operation mechanism, which is a weighted combination of numbers 1, . . . , 9.

B. In the case of another masking method, as illustrated in Figure 6, a set of distorted images is given according to relationship (6) by the matrix:

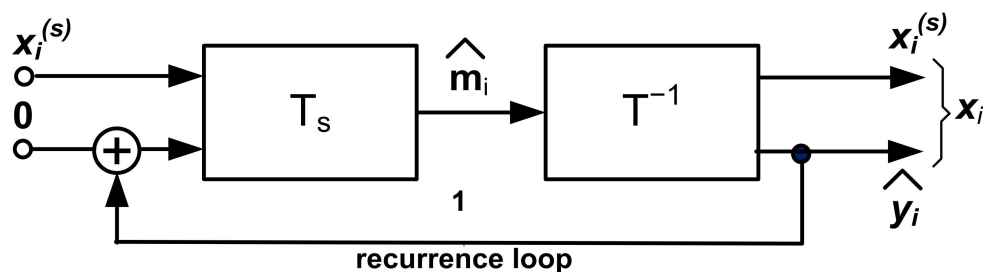
$$X^{(s)} = [x_1^{(s)}, x_2^{(s)}, \dots, x_N^{(s)}] \tag{22}$$

where  $dim x_i^{(s)} = (k \cdot n \times 1), i = 1, \dots, N, k < m$ .



**Figure 6.** Masked image of the face in Photo Number 9 (Figure 3).

The model structure for the reconstruction of such distorted images is shown in Figure 7.



**Figure 7.** Structure of the reconstruction system when a fragment of the image (k-lines) is kept as the input.

The image reconstruction process in Figure 6 is illustrated in Figure 8, which shows the results obtained after 1, 2, 5, 10, and 100 iterations. After 100 iterations, the reconstruction MSE is 0, and the identification index is 9.0. It is worth comparing the above values with the data for Photo Number 9 presented in Tables 1 and 3.

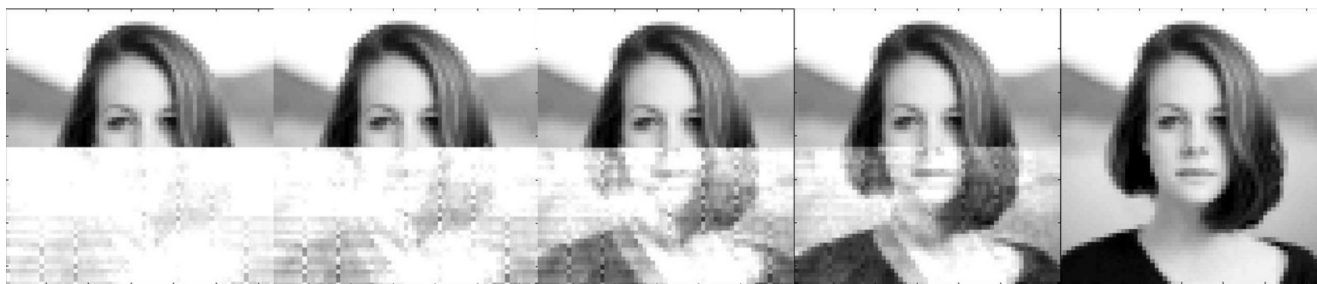


Figure 8. Image reconstruction process in Figure 6 (after 1, 2, 5, 10, and 100 iterations).

It is worth noting that, as mentioned in the Introduction, the reconstruction of Lena’s photo was realized by using the structure presented in Figure 7 [1], as well. For example, one of the distorted images of Lena and the reconstruction is shown in Figure 9.



Figure 9. Image reconstruction of Lena’s photo (reconstruction system in Figure 7).

The potential reconstruction of a distorted image using the structure in Figure 1 is illustrated for Photo Number 9 (Figure 3) by superimposing a noise vector generated by using the RAND function in MATLAB. The measure of this distortion is the signal/noise ratio expressed in decibels. The results of such a reconstruction are presented in Table 4 and Figure 10.

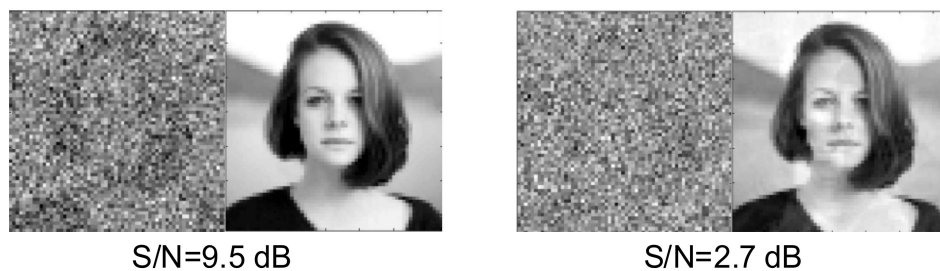


Figure 10. Reconstruction of distorted images (Items 10 and 14 in Table 4).

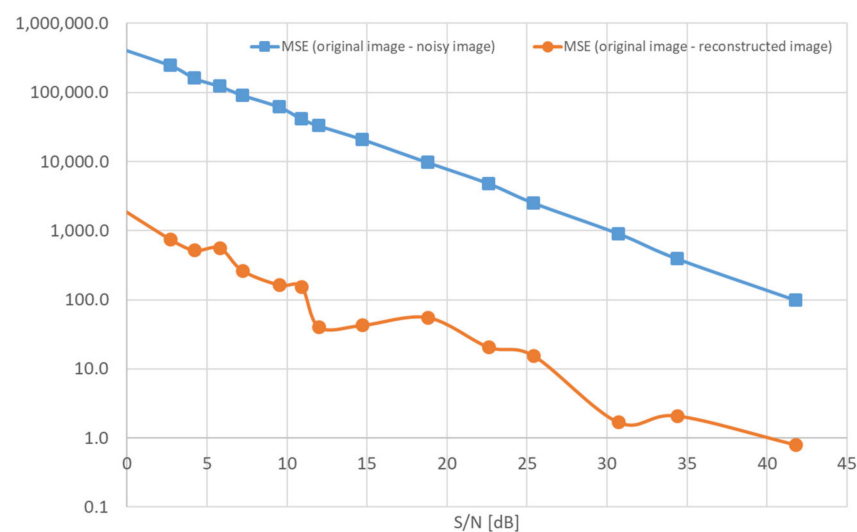


**Table 4.** Mean squared error of reconstruction.

S/N Ratio [dB]	MSE (Original Image—Noisy Image)	MSE (Original Image—Reconstructed Image)	Index Value
41.8	98.6	0.8	9.04
34.4	391.9	2.1	8.97
30.7	909.3	1.7	9.01
25.4	2523.1	15.6	8.79
22.6	4807.1	20.7	8.78
18.8	9698.9	55.6	9.39
14.7	20,942.0	42.9	8.54
12.0	32,932.0	40.4	9.06
10.9	41,500.0	155.2	9.14
9.5	61,752.0	164.5	8.66
7.2	91,527.0	266.6	9.53
5.8	122,950.0	563.4	8.13
4.2	16,0360.0	521.0	7.14
2.7	24,6230.0	754.2	10.45
−2.5	62,9540.0	4375.8	11.17

Based on Table 4, the machine learning system correctly and automatically identified the distorted image at S/N > 10 dB. Yet, even at S/N = 2.7 dB in the reconstructed image, significant similarity to the saved original photo is observed.

The numerical data in Table 4, set as a function, MSE vs. S/N, form the plots presented in Figure 11.

**Figure 11.** The plots of a function: MSE vs. S/N.

### 3. Inpainted Image Recognition and Reconstruction as an Inverse Problem

The image reconstruction models presented in the previous sections are based on the availability of training sets  $S$  in Equations (7) and (22) containing original and damaged patterns. Alternatively, a common model of image reconstruction is given by the equation:

$$Ax = \tilde{y} \quad (23)$$

where

$A$ —known processing operator, for example,  $A$  is a matrix;

$x$ —original image; and

$\tilde{y}$ —observed degenerate image.

According to Equation (23), the reconstruction of an image leads to solving the inverse problem. Most of the solutions to Equation (23) in the literature use an optimization solution [14,21], for example:

$$\min_x \|\tilde{y} - Ax\|_2^2, \quad \text{s. t. } x \in K \text{ or } \min_x \|\tilde{y} - Ax\|_2^2 + \beta R(x) \quad (24)$$

where

$K$ —set of feasible solutions;

$R(x)$ —regularizer;

$\beta$  regularization parameter.

As mentioned above, different types of neural networks are currently used to solve inverse problems in imaging, including image reconstruction. Many approaches to this problem can be found in recent reviews [22,23] and the novel proposal in [24].

The use of the machine learning model shown in Figure 1 to solve Equation (23) leads to the solution of the following problem:

$$F(x) : Ax = \tilde{y} \quad x = F^{-1}(\tilde{y}) \quad (25)$$

where

$A$ —( $m \times n$ ) known real matrix,  $m \neq n$ ;

$\tilde{y}$ —( $m \times 1$ ) real matrix;

$x$ —( $n \times 1$ ) real matrix;

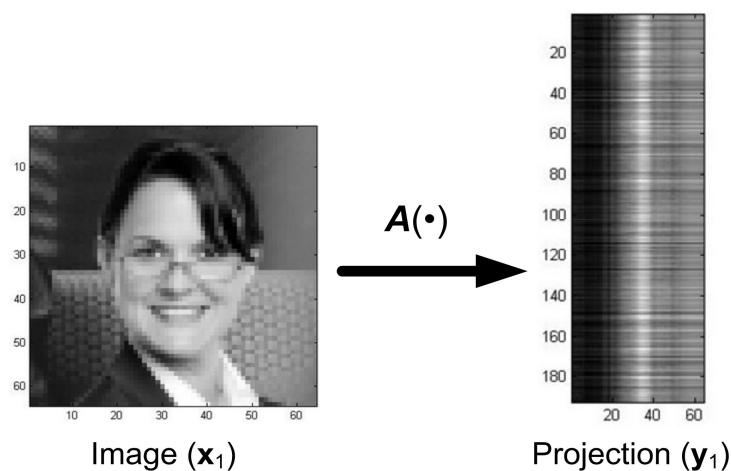
$m + n = 2^k$ ,  $k = 3, 4, \dots$

The case of  $m \neq n$  is still under consideration. The generation of the training set  $S = \{x_i, y_i\}_{i=1}^N$  for Equation (23) is given by:

$$Ax_i = y_i, \quad i = 1, 2, \dots, N \quad (26)$$

where  $x_i$ ,  $i = 1, 2, \dots, N$  is the vector form of training images, for example, those shown in Figure 3.

Assuming that the matrix  $A$  ( $m \times n$ ) in projection (25) is a random matrix, the images  $y_i$  of the training set become random vectors. For example, training image number 1 takes the form shown in Figure 12.



**Figure 12.** Original image and its transformation (projection).

Thus, the vector form of the transformation of this image (No 1) is:

$$Ax_1 = y_1 \quad (27)$$

where  $A = (m \times n), m > n$ .

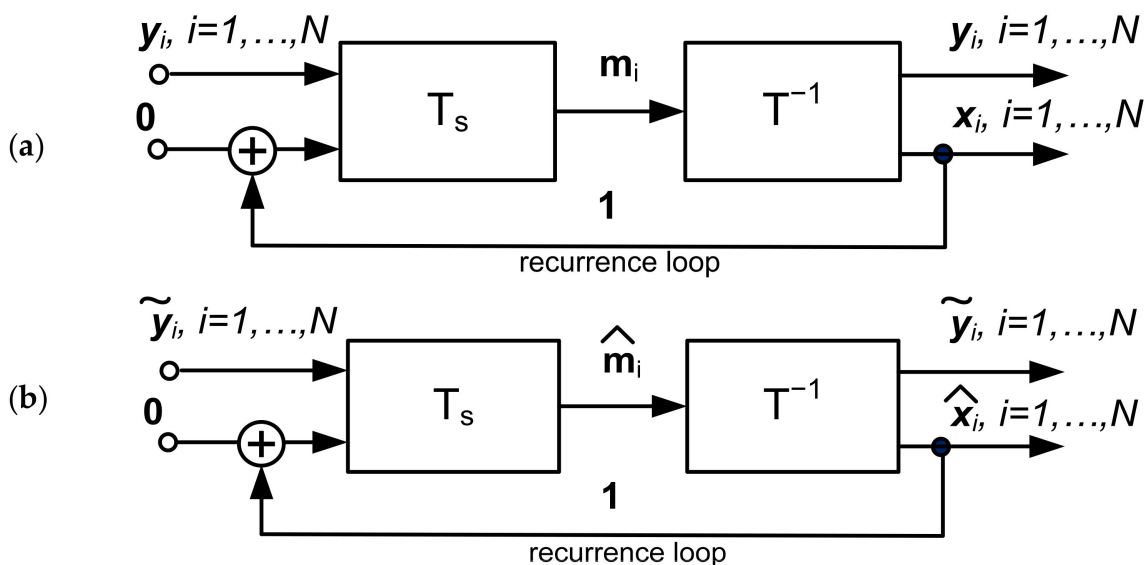
Taking the system vectors  $u_i$  of the form

$$u_i = \begin{bmatrix} y_i \\ x_i \end{bmatrix}, i = 1, \dots, N, \tag{28}$$

the structure of the inverse mapping system (25), i.e.,:

$$x_i = F^{-1}(y_i), i = 1, \dots, N \tag{29}$$

is given in Figure 13a,b.



$\hat{x}_i$ —estimator of  $x_i$

$\tilde{y}_i$ —projection of the degenerated image of the original  $x_i$ .

**Figure 13.** Structure of the system implementing inverse transformation. (a)  $y_i$ —undegenerated image projection; (b)  $\tilde{y}_i$ —degenerated image projection.

It should be noted that the biorthogonal transformation  $T_s(\cdot)$  and orthogonal transformation  $T(\cdot)$  in Figure 13 are given by Equations (16) and (17), respectively:

Thus,

$$m_i = T_s \left( \begin{bmatrix} y_i \\ 0 \end{bmatrix} \right) \tag{30}$$

$$u_i = T^{-1}(m_i) \tag{31}$$

where  $u_i$ —system vectors Equation (28).

In the system presented in Figure 13, the distorted projections of the images  $\tilde{y}_i, i = 1, \dots, N$  undergo reconstruction, in contrast with the system in Figure 7, where the distorted images are reconstructed. To illustrate the properties of the reconstruction system presented in Figure 13, a training set S was generated using Equation (26) consisting of nine images  $x_i, i = 1, \dots, 9$ , where  $x_i$  were images from Figure 3, and their projections  $y_i, i = 1, \dots, 9$  were obtained with the random matrix A. An exemplary transformation of Image Number 9 from Figure 3 is shown in Figure 14.

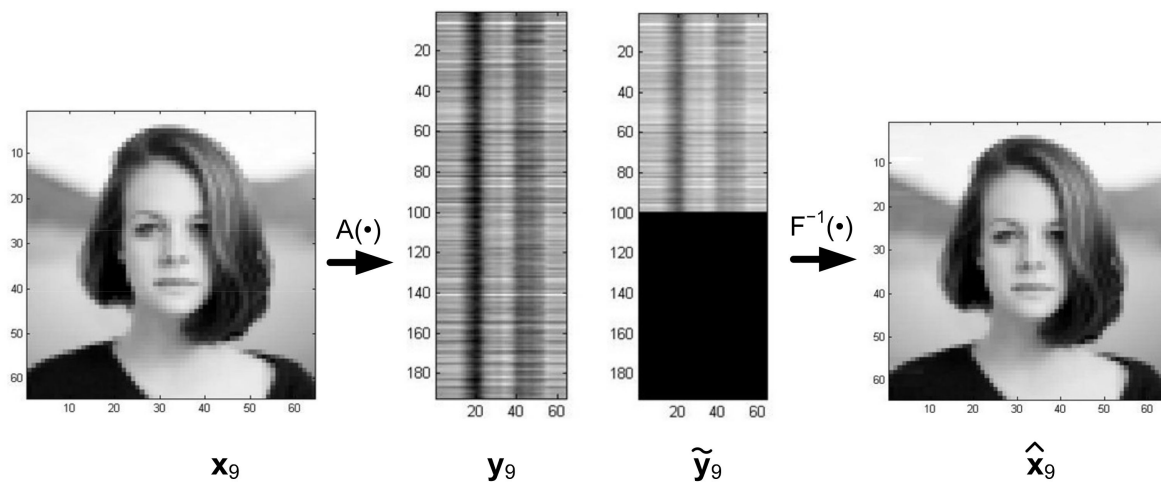


Figure 14. An exemplary reconstruction ( $F(\cdot)$ —system from Figure 13b).

In the system shown in Figure 13b, we obtain:

$$\|x_9 - \hat{x}_9\|_2^2 = 0. \tag{32}$$

To conclude, Figure 1, Figure 7, and Figure 11 show image reconstruction systems that substantially implement an associative memory structure for recognizing damaged key patterns. However, it is worth noting that on other hand, the system in Figure 13 implements inverse transformation and solves optimization tasks constrained by images stored in memory. Moreover, this system enables the solving of linear Equation (23) by using a random form of training vectors  $x_i$  in Equation (26) [1] as well.

#### 4. Discussion on Some Features of the Machine Learning System

A. This section focuses on some of the features that underlie the universality of the machine learning system presented in Figures 1 and 7. First of all, it is clear that this machine learning system can be categorized as an iterative scheme. On other hand, the structure in Figure 1 can be treated as a feedforward block connection constituting a multilayer, deep learning architecture (Figure 15).

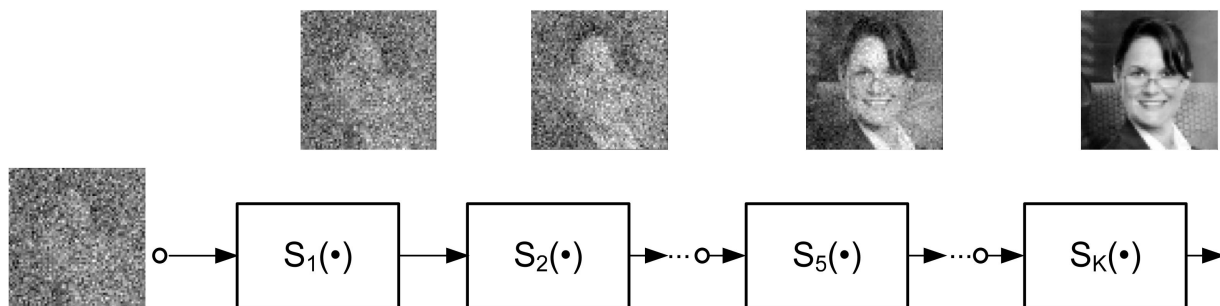


Figure 15. Multilayer learning structure ( $K$ —number of steps; e.g.,  $K = 100$ ).

The structure in Figure 7 can be similarly treated as a feedforward scheme, as shown in Figure 16.

The blocks  $S_i(\cdot)$ ,  $i = 1, \dots, K(L)$  are identical in multilayer structures (Figures 15 and 16).

It is worth nothing that the multilayer structures in Figures 15 and 16 can be seen as an implementation of deep learning using recurrent neural networks (RNN) [25]. However, the topology of these structures is not a result of optimization algorithms typically used to solve the inverse problems.

B. An interesting property of the structure in Figure 1 can be set up by a computational experiment illustrated in Figure 17; i.e., when this structure memorizes only one image (e.g., photo No 2 in Figure 3), then any image is mapped on this memorized image (a property of global attractor).

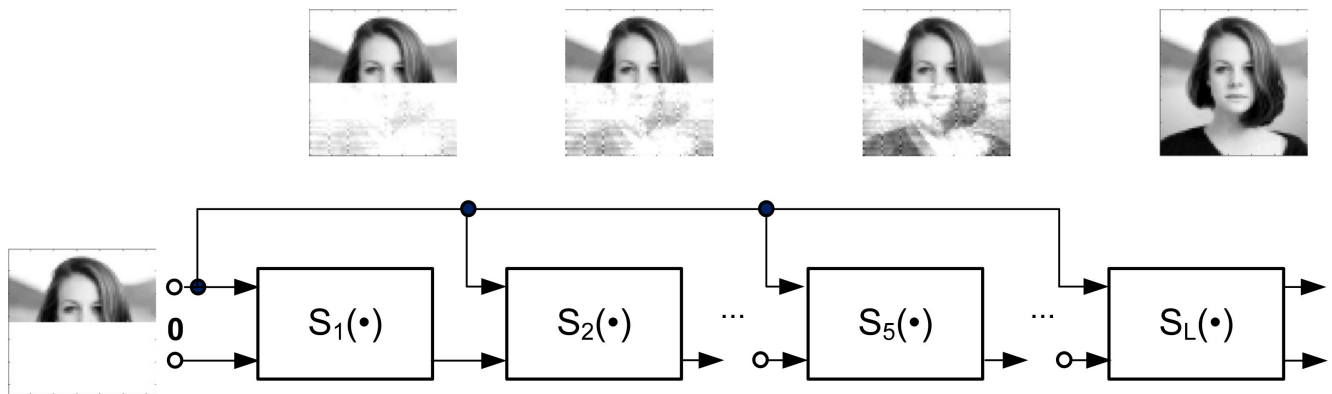


Figure 16. Multilayer learning structure ( $L$ —number of steps; e.g.,  $L = 100$ ).

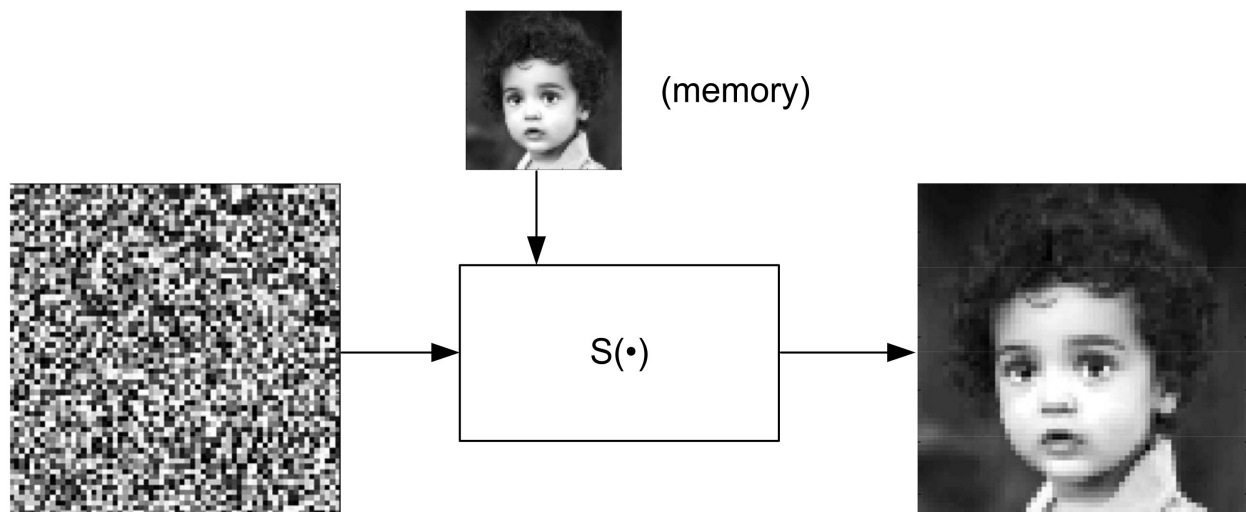
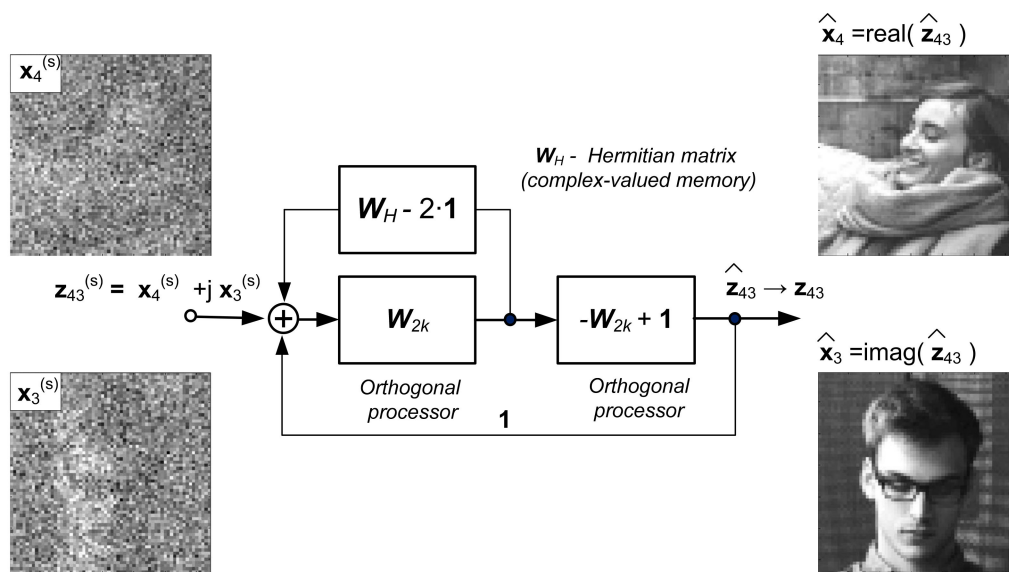


Figure 17. Illustration of global attractor properties.

C. Another interesting aspect of this machine learning system can be derived from the so-called Q-inspired neural networks feature [1]. This feature can be determined by the following statement:

Given a set of complex-valued training vectors  $\{x_i, y_i\}_{i=1}^N$  where  $x_i \in C^n$ ,  $y_i \in C^m$ ,  $n + m = 2^k$ ,  $k = 3, 4, \dots$ , a realization of mapping given by complex training vectors, i.e.,  $C^n \rightarrow C^m$  can be implemented as complex-valued neural networks or as a complex-valued machine learning system with the structure presented in Figure 2, where the memory block is determined by the Hermitian matrix  $W_H$  ( $W_s \rightarrow W_H$  in Equation (3)).

Such a machine learning system can be used as an image processor to reconstruct complex-valued images. It is clear that the computational efficiency of this system is greater than that of the real-valued approximator (due to the processing of two images by only one system). Figure 18 provides an example of complex-valued image reconstruction.



**Figure 18.** Complex-valued image reconstruction:  $z_{43} = x_4 + jx_3$ ,  $j^2 = -1$ ,  $x_3, x_4$ —vectorized form of images No. 3 and No. 4 in Figure 3,  $x_3^{(s)}, x_4^{(s)}$ —distorted images.

D. This article focus on image processing by using the recursive machine learning system in Figure 1. It should be clear that this image processing sets up only one aspect of the potential system applicability in the field of signal processing. For example, the same machine learning could be used for time-series analysis and forecasting. To generalize, the essential function of the machine learning system described in this paper is the implementation of mapping defined by a training set  $S = \{x_i, y_i\}_{i=1}^N$ ;  $i = 1, \dots, N$ , where  $\dim x_i = n$ ,  $\dim y_i = m$ . The recurrence is convergent under the linear independence of input vectors, and the number of vectors  $N$  fulfills (see Equation (5))  $N < 0.5(n + m)$ ,  $n + m = 2^k$ ,  $k = 2, 4, \dots$ . Thus, a large capacity system (large  $N$ ) needs a large even dimension ( $n + m$ ) system. It could be considered as a disadvantage of this machine learning systems.

## 5. Conclusions

The aim of this article was to illustrate the potential for using the machine learning system shown in Figure 1 to reconstruct and recognize distorted or damaged patterns, in particular, images of people wearing masks. In contrast to the image reconstruction methods based on using optimization algorithms, this system employs the superposition of system vectors setting up asymptotic centers of attraction. Hence, this system is particularly useful for the implementation of associative memories. Thus, this paper's inpainted image reconstruction sets up associative memories where images stored in memory are retrieved by distorted/inpainted key images. To conclude, we formulated another image processing tool augmenting the set of known image processing methods. Finally, all the image reconstructions presented in this paper, were done using MATLAB (The Math Works, Inc. MATLAB version 2021b).

**Author Contributions:** Conceptualization, W.S. and W.C.; methodology, W.S.; software, W.C.; validation, W.S. and W.C.; formal analysis, W.S.; investigation, W.S.; resources, W.S.; data curation, W.C.; writing—original draft preparation, W.C.; writing—review and editing, W.C. and W.S.; visualization, W.C.; supervision, W.S.; project administration, W.C.; funding acquisition, W.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Ministry of Science and Higher Education through the “Regionalna Inicjatywa Doskonalosci” Program in 2019–2022 under Project 006/RID/2018/19, and the total financing is 11 870 000 PLN.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A

---

**Algorithm A1.** Summary of algorithm [1].

---

1 **Declaration:**

Input the set of training points:

$$S = \{x_i, y_i\}, i = 1, 2, \dots, N,$$

$$x_i \in R^n, y_i \in R^m, n + m = 2^k, k = 3, 4, \dots$$

2 **System design**

Create system vectors  $u_i$  :

$$u_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}, \dim u_i = n + m.$$

Calculate the spectrum  $m_i$  of system vectors  $u_i$  :

$$m_i = \frac{1}{2}(W_{2^k} + \mathbf{1})u_i$$

Create spectrum matrix  $M$  :

$$M = [m_1, m_2, \dots, m_N]$$

Calculate Hermitian matrix  $W_H$ :

$$W_H = M(M^T M)^{-1} M^T$$

Calculate orthogonal transformation  $T(\cdot)$ :

$$T(\cdot) \equiv T = \frac{1}{2}(W_{2^k} + \mathbf{1})$$

Calculate biorthogonal transformation  $T_s(\cdot)$ :

$$T_s(\cdot) \equiv T_s = (2 \cdot \mathbf{1} - W_H - W_{2^k})^{-1}.$$

3 **Recursive procedure:**

for  $i = 1:N$

$$\tilde{x}_i^{(0)} = \mathbf{0}$$

while  $\|\tilde{x}_i^{(l)} - \tilde{x}_i^{(l-1)}\| \geq eps$

$$\begin{bmatrix} \tilde{x}_i \\ y_i \end{bmatrix}^{(l)} = T^{-1} T_s \left( \begin{bmatrix} \mathbf{0} \\ y_i \end{bmatrix} + \begin{bmatrix} \tilde{x}_i \\ \mathbf{0} \end{bmatrix}^{(l-1)} \right)$$

end

end

( $l = 1, 2, \dots$  steps of recurrence)

---

Final results of recurrence :  $\tilde{x}_i = x_i$ .

---

## References

1. Citko, W.; Sienko, W. Hamiltonian and Q-Inspired Neural Network-Based Machine Learning. *IEEE Access* **2020**, *8*, 220437–220449. [[CrossRef](#)]
2. Gonzales, R.C.; Woods, R.E. *Digital Image Processing*; Pearson International Edition; Pearson: London, UK, 2008.
3. Nelson, R.A.; Roberts, R.G. Some Multilinear Variants of Principal Component Analysis: Examples in Grayscale Image Recognition and Reconstruction. *IEEE Syst. Man Cybern. Mag.* **2021**, *7*, 25–35. [[CrossRef](#)]
4. Sirovich, L.; Kirby, M. Low Dimensional Procedure for the Characterization of Human Faces. *J. Opt. Soc. Am.* **1987**, *4*, 519–524. [[CrossRef](#)] [[PubMed](#)]
5. Turk, M.; Pentland, A. Face Recognition Using Eigenfaces. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '91), Maui, HI, USA, 3–6 June 1991; pp. 586–591.
6. Pal, S.K.; Ghosh, A.; Kundu, M.K. (Eds.) *Soft Computing for Image Processing*. In *Studies in Fuzziness and Soft Computing*; Physica-Verlag Heidelberg: New York, NY, USA, 2000.
7. Huang, Z.; Ye, S.; McCann, M.T.; Ravishankar, S. Model-based Reconstruction with Learning: From Unsupervised to Supervised and Beyond. *arXiv* **2021**, arXiv:2103.14528v1.

8. Kaderuppan, S.S.; Wong, W.W.L.; Sharma, A.; Woo, W.L. Smart Nanoscopy: A Review of Computational Approaches to Achieve Super-Resolved Optical Microscopy. *IEEE Access* **2020**, *8*, 214801–214831. [[CrossRef](#)]
9. Ramanarayanan, S.; Murugesan, B.; Ram, K.; Sivaprakasam, M. DC-WCNN: A Deep Cascade of Wavelet Based Convolutional Neural Networks for MR Image Reconstruction. In Proceedings of the IEEE 17th International Symposium on Biomedical Imaging (ISBI), Iowa City, IA, USA, 3–7 April 2020.
10. Ravishankar, S.; Lahiri, A.; Blocker, C.; Fessler, J.A. Deep Dictionary-transform Learning for Image Reconstruction. In Proceedings of the 2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018), Washington, DC, USA, 4–7 April 2018; pp. 1208–1212. [[CrossRef](#)]
11. Ravishankar, S.; Ye, J.C.; Fessler, J.A. Image Reconstruction: From Sparsity to Data-Adaptive Methods and Machine Learning. *Proc. IEEE* **2020**, *108*, 86–109. [[CrossRef](#)] [[PubMed](#)]
12. Ravishankar, S.; Bresler, Y. MR Image Reconstruction From Highly Undersampled k-Space Data by Dictionary Learning. *IEEE Trans. Med. Imaging* **2021**, *30*, 1028–1041. [[CrossRef](#)] [[PubMed](#)]
13. Panagakis, Y.; Kossaifi, J.; Chrysos, G.G.; Oldfield, J.; Nicolaou, M.A.; Anandkumar, A.; Zafeiriou, S. Tensor Methods in Computer Vision and Deep Learning. *Proc. IEEE* **2021**, *109*, 863–890. [[CrossRef](#)]
14. Fessler, A.J. Optimization Methods for Magnetic Resonance Image Reconstruction: Key Models and Optimization Algorithms. *IEEE Signal Process. Mag.* **2020**, *37*, 33–40. [[CrossRef](#)] [[PubMed](#)]
15. Zheng, H.; Sherazi, S.W.A.; Son, S.H.; Lee, J.Y. A Deep Convolutional Neural Network-Based Multi-Class Image Classification for Automatic Wafer Map Failure Recognition in Semiconductor Manufacturing. *Appl. Sci.* **2021**, *11*, 9769. [[CrossRef](#)]
16. Szhou, S.K.; Greenspan, H.; Davatzikos, C.; Duncan, J.S.; Ginneken, B.; Madabhushi, A.; Prince, J.L.; Rueckert, D.; Summers, R.M. A Review of Deep Learning in Medical Imaging: Imaging Traits, Technology Trends, Case Studies with Progress Highlights, and Future Promise. *Proc. IEEE* **2021**, *109*, 820–838.
17. Quan, T.M.; Nguyen-Duc, T.; Jeong, W.K. Compressed Sensing MRI Reconstruction Using a Generative Adversarial Network with a Cyclic Loss. *IEEE Trans. Med. Imaging* **2018**, *37*, 1488–1497. [[CrossRef](#)] [[PubMed](#)]
18. Mardani, M.; Gong, E.; Cheng, J.Y.; Vasanawala, S.S.; Zaharchuk, G.; Xing, L.; Pauly, J.M. Deep Generative Adversarial Neural Networks for compressive sensing MRI. *IEEE Trans. Med. Imaging* **2019**, *38*, 167–179. [[CrossRef](#)] [[PubMed](#)]
19. Liang, D.; Cheng, J.; Ke, Z.; Ying, L. Deep Magnetic Resonance Image Reconstruction: Inverse Problems Meet Neural Networks. *IEEE Signal Process. Mag.* **2020**, *37*, 141–151. [[CrossRef](#)] [[PubMed](#)]
20. Sienko, W.; Citko, W. Hamiltonian Neural Networks Based Networks for Learning. In *Machine Learning*; Mellouk, A., Chebira, A., Eds.; I-Tech: Vienna, Austria, 2009; pp. 75–92.
21. Gilton, D.; Ongie, G.; Willett, R. Deep Equilibrium Architectures for Inverse Problems in Imaging. *arXiv* **2021**, arXiv:2102.07944v2. [[CrossRef](#)]
22. Arridge, S.; Maass, P.; Oktem, O.; Schonlieb, C. Solving Inverse Problems using Data-driven Models. *Acta Numer.* **2019**, *28*, 1–174. [[CrossRef](#)]
23. Ongie, G.; Jalal, A.; Metzler, C.A.; Baraniuk, R.G.; Dimakis, A.G.; Willett, R. Deep Learning Techniques for Inverse Problems in Imaging. *arXiv* **2020**, arXiv:205.06001. [[CrossRef](#)]
24. Gilton, D.; Ongie, G.; Willett, R. Model Adaptation for Inverse Problems in Imaging. *arXiv* **2021**, arXiv:2012.00139v2. [[CrossRef](#)]
25. Giryes, R.; Eldar, Y.C.; Bronstein, A.M.; Sapiro, G. Tradeoffs Between Convergences Speed and Reconstruction Accuracy in Inverse Problems. *arXiv* **2018**, arXiv:1605.09232v3. [[CrossRef](#)]