



Image interpolation with spiking neural network based pixel similarity

Mahmut Kılıçaslan¹

Received: 23 April 2024 / Revised: 31 May 2024 / Accepted: 6 June 2024
© The Author(s) 2024

Abstract

Image interpolation is an important topic in the field of image processing. It is defined as the process of transforming low-resolution images into high-resolution ones using image processing methods. Recent studies on interpolation have shown that researchers are focusing on successful interpolation techniques that preserve edge information. Therefore, the edge detection phase plays a vital role in interpolation studies. However, these approaches typically rely on gradient-based linear computations for edge detection. On the other hand, non-linear structures that effectively simulate the human visual system have gained attention. In this study, a non-linear method was developed to detect edge information using a pixel similarity approach. Pixel similarity-based edge detection approach offers both lower computational complexity and more successful interpolation results compared to gradient-based approaches. 1D cubic interpolation was applied to the pixels identified as edges based on pixel similarity, while bicubic interpolation was applied to the remaining pixels. The algorithm was tested on 12 commonly used images and compared with various interpolation techniques. The results were evaluated using metrics such as SSIM and PSNR, as well as visual assessment. The experimental findings clearly demonstrated that the proposed method outperformed other approaches. Additionally, the method offers significant advantages, such as not requiring any parameters and having competitive computational cost.

Keyword Interpolation · Pixel similarity · Edge preservation

1 Introduction

With the rapid development of networking and communication technologies, multimedia applications such as image and video have become increasingly widespread. As a result, techniques developed for these environments have become the focus of researchers. While high-resolution images can be obtained with hard-ware, it is a costly process. Additionally, the unavailability of image sources in the desired dimensions presents another challenge. Therefore, soft-ware solutions for image interpolation are indispensable [1]. Image interpolation, which is based on generating a high-resolution image from a low-resolution image, is a fundamental research topic frequently employed in image processing. It finds applications in various fields such as digital photography, biomedical imaging, remote sensing, and forensic analysis [2–4]. When the literature is reviewed, there are many methods developed for image interpolation. These interpolation

techniques can be categorized into two groups: learning-based and traditional (reconstruction-based) methods [5]. Learning-based methods [6–11] utilize training data, extracting information from training examples to generate new pixels. Traditional methods [12–17], on the other hand, rely on the original image to formulate interpolation functions. Moreover, they perform the interpolation process through mathematical formulas without requiring any training step. The most well-known mathematical approaches are nearest neighbor, bilinear, and bicubic interpolation techniques [12–14]. These methods are simple to implement and have low computational costs. However, they tend to introduce blurring and distortions at the edges. This is primarily due to the fact that these methods do not consider any information other than the intensity values [15–17]. Consequently, edge-based image interpolation approaches have gained attention in recent years [18, 19]. These methods are designed to adapt to the human visual system and prioritize edge information in the image [20–24].

IEDI [25], CGI [26], CED [27], PCI [28], WTCGI [29], and GEI [30] are some of the edge-based interpolation techniques proposed in recent years. In the IEDI [25] technique,

✉ Mahmut Kılıçaslan
m.kilicaslan@ankara.edu.tr

¹ Department of Computer Technologies, Ankara University, Ankara, Turkey

the edge pixels of the images are detected using an adaptive edge filter. The main drawback of the IEDI method is its high computational time. Contrast-Guided Interpolation (CGI) [26], proposed in 2013, is another well-known interpolation approach. The authors utilized the contrast feature, which is important for the human visual system. Additionally, in the study, the interpolation process was performed in collaboration with guided decision maps and edge filtering algorithms. However, while the CGI approach produces successful results, its iterative nature results in high computational cost. Another effective approach is the CED [27] technique, which is an improved version of the CGI approach. This interpolation algorithm maintains the performance of the CGI approach while increasing the time and computational cost of diffusion processes. The PCI [28] approach, proposed in 2019, uses another version of the CGI technique. In this approach, a prediction-validation model is employed. The WTCGI [29] technique combines wavelet transform and CGI structure and is applied to the interpolation of images used in the robotics industry. The GEI [30] technique, developed in 2022, performs regional gradient estimation. This approach has shown successful results. On the other hand, an approach proposed by İncetaş in 2023 is a significant study where edges are determined nonlinearly [31]. The author utilized the well-known spike neural network (SNN) architecture, which successfully simulates the human visual system. In relevant study, edge pixels were calculated nonlinearly using the presented SNN model instead of a linear approach like gradient.

The general motivation behind the aforementioned interpolation approaches is to detect edges by using gradient-based linear methods. On the other hand, it is well-known that the human visual system is more effectively simulated using nonlinear models [32, 33]. In 2007, Demirci developed a similarity-based nonlinear edge detection approach. In the proposed approach for edge detection in color images, the similarity relationship matrix mapped the 3D color space to a one-dimensional space. The developed approach served as an alternative to gradient and statistical models. Additionally, its faster and more efficient operation is an important feature of the method [34]. Pixel similarity has been a popular approach used in many studies, such as edge detection [35], segmentation [36, 37], and noise filters [38]. In a study conducted in 2017, pixel similarity was applied to 200 images from the BSDS database. Threshold values were computed based on the similarity images to discuss the effect of pixel similarity. In another study, the relevant method was used for the automatic separation of color images [36]. The researchers calculated the number of regions in their work without the need for any initial value. Similarly, there are various studies demonstrating the effective results of the pixel similarity approach in image segmentation [37]. Additionally, Aydın et al. applied the pixel similarity approach to medical images.

They utilized advanced image processing techniques for the early diagnosis of jaundice instead of blood samples and clinical tests. In their studies, they proposed a prediction model with high accuracy using the pixel similarity approach [39]. Another significant study is the proposal of using pixel similarity as a noise filter [38]. In [35], the distance values of pixels to their neighbors were evaluated as the diffusion coefficient. Thus, the traditional user-dependent diffusion filter was made adaptive. Pixel similarity shows effective performance in edge detection, it is based on linear calculation of the difference between pixels. On the other hand, the SNN-based model, which detects edges in accordance with HVS proposed in 2023, appears to be quite successful [31]. However, that study presented a complex model in which calculations had to be made on all 9 pixels in the 3×3 window at each stage when determining the edges. In this study, a novel approach is proposed to increase the achievement of image interpolation and edge preservation ability and reduce the calculation time by using SNN-based similarity transformation. In this study, a novel approach is proposed to increase the achievement of image interpolation and edge preservation ability by using SNN-based similarity transform. In the proposed interpolation approach, firstly, the interpolation type is selected according to the edge information obtained non-linearly from the SNN based pixel similarity of the image. While 2D-Bicubic interpolation is performed on non-edge pixels, 1D interpolation is performed on edge pixels. Thus, while the edge losses have been minimized and the interpolation success has been increased. Edge information was detected more accurately with the SNN-based pixel similarity approach. The main contribution of this work over previously approaches is the introduction of an SNN-based pixel similarity model compatible with HVS. When determining edge pixels, the gray level differences of the pixels are calculated non-linearly, unlike gradient-based linear approaches. A simpler and faster SNN model than existing SNN models is presented.

The organization of the following sections of the article is as follows; Chapter 2 Image Interpolation, Chapter 3 SNN based Pixel Similarity, Chapter 4 Proposed Method, Chapter 5 Experimental Findings and Discussion, and finally Results.

2 Image interpolation

Image interpolation refers to the process of generating a high-resolution (*HR*) image from a low-resolution (*LR*) image and is an important topic in image processing. Figure 1 represents the creation of an image with a resolution of $2M \times 2N$ from an original image with a resolution of $M \times N$. In the resulting image, the unknown pixels (white dots with red lines) are estimated using the known pixels (blue dots). The estimation

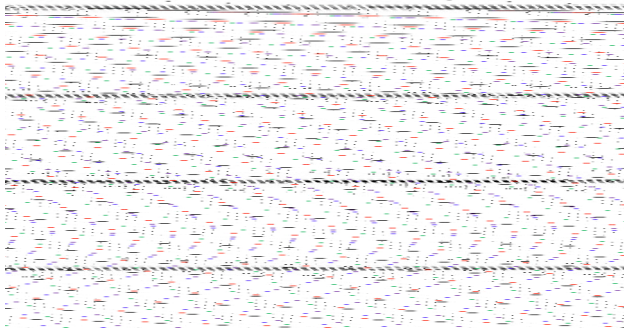


Fig. 1 Scheme of interpolation

$p1$	$p2$	$p3$
$p8$	$p0$	$p4$
$p7$	$p6$	$p5$

Fig. 2 Center pixel and its neighborhood

of the unknown pixels in the interpolation process is a critical operation. In Fig. 1, the low-resolution image is denoted as $I_{LR}(m, n)$, while the high-resolution image obtained through 2×2 image interpolation is represented as $I_{HR}(2m-1, 2n-1)$. The relationship between I_{LR} and I_{HR} is shown in Eq. 1, where the known pixel values in I_{LR} are assigned to I_{HR} . Then, the detection of edges and their orientations is performed. The estimation of unknown pixel values starts with the pixels on the diagonal axis which $I_{HR}(2m, 2n)$. If the pixel at $I_{LR}(m, n)$ is an edge, its value is calculated using the known neighboring pixel values. The next step is to compute the values of unknown pixels in the horizontal and vertical positions. The unknown pixel values at $I_{HR}(2m, 2n+1)$ in the horizontal position and $I_{HR}(2m+1, 2n)$ in the vertical position are also calculated using the known neighboring pixels. Finally, all the unknown pixels in I_{HR} are computed as:

$$\begin{aligned} I_{LR}(m, n) &= I_{HR}(2m-1, 2n-1), \quad m \\ &= 1, \dots, M, \quad n = 1, \dots, N \end{aligned} \quad (1)$$

3 Spike neural network based pixel similarity

The pixel similarity approach was proposed by Demirci [34]. The developed approach is a pixel similarity-based edge detection technique. The notable aspect of this technique is the absence of complex steps. The pixel similarity technique is essentially based on mathematically expressing the similarity between two pixels. Figure 2 illustrates a 3×3 pixel

block, where $p0$ represents the center pixel and $p1, p2, \dots, p8$ are its neighbors. The similarity of the center pixel ($p0$) to any other pixel (pi) can be mathematically expressed as;

$$\text{Similarity}(p0, pi) = e^{-\frac{\|p0-pi\|}{Dn}} \quad (2)$$

$$\|p0 - pi\| = \frac{1}{\sqrt{3}} \sqrt{(\Delta R^2 + \Delta G^2 + \Delta B^2)} \quad (3)$$

$$\begin{aligned} \Delta R &= |R0 - Ri| \\ \Delta G &= |G0 - Gi| \\ \Delta B &= |B0 - Bi| \end{aligned} \quad (4)$$

$\|p0 - pi\|$ defines the normalized Euclidean distance between two pixels in Eq. 2. Dn is the normalization coefficient and the relevant value is determined by the user. The average similarity (A_{sim}) of a 3×3 mask is formulated as:

$$A_{sim}(p0) = \frac{\sum_{i=0}^8 \text{similarity}(p0, pi)}{9} \quad (5)$$

The outputs with similarity change graph according to the different Dn coefficient determined by the user is shown in Fig. 3. In addition, the output images of a randomly selected image with pixel similarity applied according to the relevant Dn coefficients are represented in Fig. 4.

The non-linear structure of the similarity function in Fig. 3 is remarkable. When Fig. 4 is examined, it is seen that the pixel values approach 255 with the increase in the Dn coefficient. Therefore, the detection of correct edge pixels depends on the appropriate selection of the normalization coefficient Dn . Incetaş et al. in their study in 2019, made the relevant coefficient independent of the user. As a result, they developed an innovative automatic edge detection model in their work. Edge pixels detected with the automatically detected Dn coefficient produced results similar to the conventional method [40]. However, it is important that the relevant coefficient is user independent. Equations 6 and 7 show how the Dn is calculated automatically.

$$Dn = \left(\frac{255}{d_a^2 + 1} \right) + 1 \quad (6)$$

$$d_a = \frac{1}{9} \sum_{j=0}^8 d_{0,j} \quad 0 \leq j \leq 8 \quad (7)$$

where d_a represents the average distance between the center pixel and its neighbors and $d_{0,j}$ is the similarity of any neighboring pixels. Figure 5 is the output obtained by pixel similarity of a cameraman's image obtained by automatically determining the Dn coefficient.

While pixel similarity produces effective outcomes for edge detection, it is based on simple calculations between

Fig. 3 Pixel similarity graph according to Eq. (2)

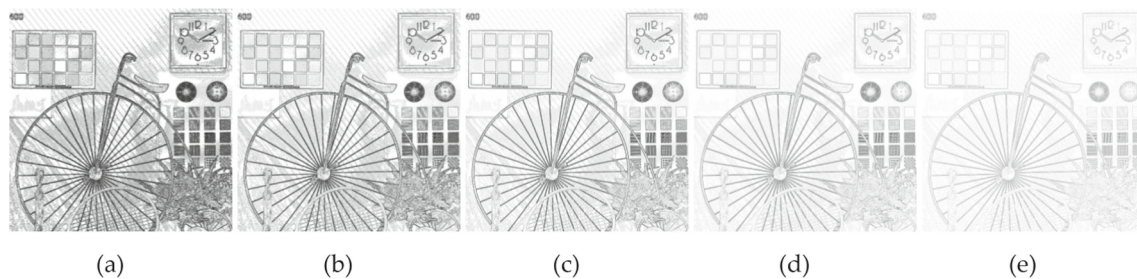
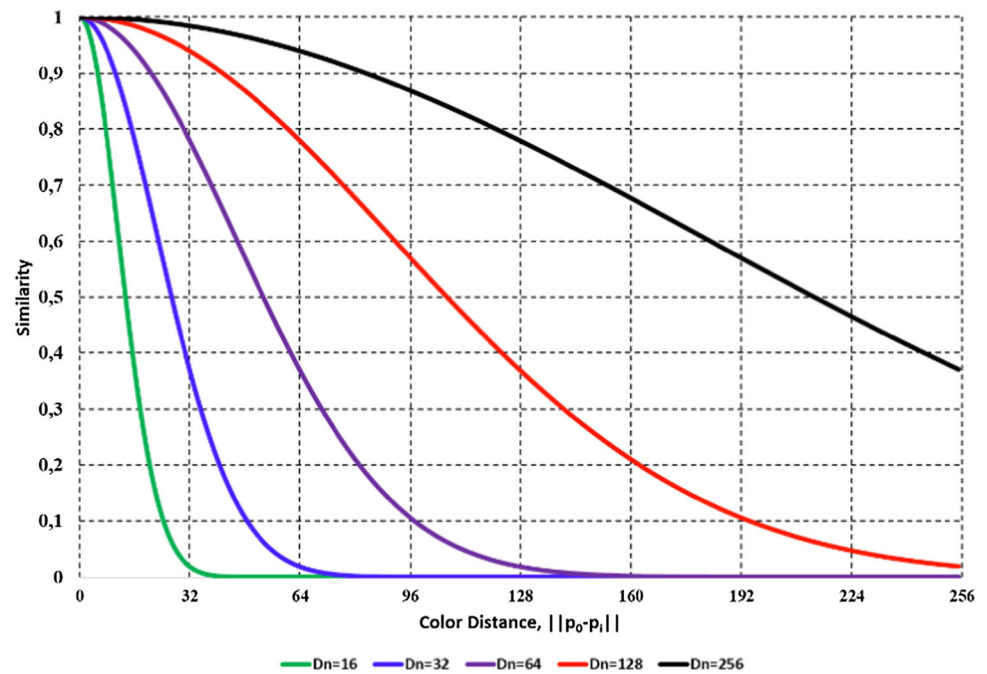


Fig. 4 Similarity images: Wheel **a** $Dn = 16$, **b** $Dn = 32$, **c** $Dn = 64$, **d** $Dn = 128$, **e** $Dn = 256$

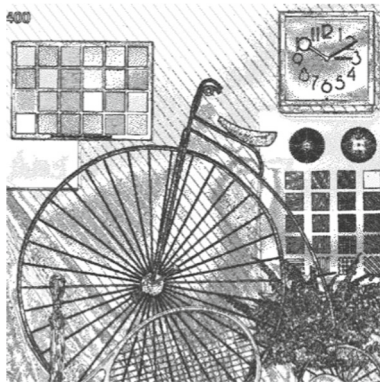


Fig. 5 Pixel similarity with automatic Dn coefficient

two pixels. Therefore, it produces similar results to gradient-based approaches. Because of this situation, the pixel similarity technique considers the absolute difference or gradient difference between two-pixel values. In other words, although pixel similarity seems to be a non-linear approach,

it is actually linear because it calculates the absolute differences between pixels. On the other hand, the SNN approach is a network approach that effectively simulates the human visual system. In the SNN approach, the difference values between two pixels are calculated by the firing rates of the spikes generated depending on the membrane voltage of the neurons modeled by the conductance based integrate and fire.

In the early 1950s, a study was proposed by Hodgkin and Huxley (HH) to define the characteristics of human neurons. In their study, a series of mathematical processes were used to describe the functions of neurons. In other words, neuron behaviors were simulated with differential equations [41]. The presence of complex mathematical solutions in HH's successful approach increases its computational cost. For this reason, different neuron models such as integrate-and-fire (IF), FitzHugh–Nagumo (FHN) and Izhikevich techniques have been presented [42–44]. The IF neuron model attracts attention because it has a simpler architecture and lower computational complexity in comprehensive neural systems. In the IF neuron model, change of membrane potential with

time $v(t)$ [45] is formulated as follows:

$$c_m \frac{dv(t)}{dt} = g_l(E_l - v(t)) + \frac{w_{ex}g_{ex}(t)}{A_{ex}}(E_{ex} - v(t)) + \frac{w_{ih}g_{ih}(t)}{A_{ih}}(E_{ih} - v(t)) \quad (8)$$

Here c_m , g_l and E_l represent the membrane capacity, membrane conductance and membrane reversal potential, respectively. E_{ex} stands for the reversal potential of excitatory synapses, while E_{ih} stands for the reversal potential of inhibitory synapses. w_{ex} and w_{ih} are the weights of the synapses and A_{ex} and A_{ih} are the membrane surface areas. The variable $g_{ex}(t)$ is the time-varying conductance of excitatory synapses. Similarly, $g_{ih}(t)$ is the time-varying conductance of inhibitory synapses. The excitatory connections of a neuron can send a higher signal than the inhibitory connections. In this case, the membrane potential of the neuron reaches a value above the threshold voltage v_{th} at time t and the neuron generates a new action potential. Subsequently, starting at time $t + 1$, the membrane potential of the neuron remains constant at its initial value v_{reset} for a period of time τ_{ref} , called the refractory time. The value of τ_{ref} is important for the ease of calculation of the model. For this reason, in this study, the relevant value is set to 0 as in similar studies [46–48]. Figure 6 shows a basic IF neuron model.

The proposed study is introduced a pioneering edge detection methodology employing Spiking Neural Network (SNN) architecture, which effectively compete with the Human Visual System (HVS). Unlike traditional linear edge detection models relying on pixel similarity, this novel approach incorporates a multi-layered SNN structure depicted in Fig. 7. The SNN comprises three layers: the receptor layer, an intermediate layer, and an output layer. In the receptor layer, each pixel in the image is represented, forming a 3×3 receptor field (RF). The relationships between the central pixel and its neighbors are analyzed through synaptic connections to the intermediate layer, which consists of 16 separate neurons facilitating computations in 8 directions. These neurons measure the difference between the central pixel and its neighboring pixels using excitatory and inhibitory synapses represented by synaptic matrices denoted as X and Δ respectively. The intermediate layer neurons are interconnected with 8 separate output neurons to express the similarities between the central pixel and its neighbors. The firing frequency of each output neuron indicates the dissimilarities between the central pixel and its neighbors. For instance, the synaptic connections from the RF to a specific neuron in the intermediate layer which M_{5A} , facilitate computations between the central pixel and its bottom neighbor. Excitatory synapses from the neighbor to M_{5A} and inhibitory synapses from the central pixel to M_{5A} modulate the membrane potential of M_{5A} . An increase in the

membrane potential of M_{5A} shows a higher gray-level value in the neighbor compared to the central pixel, leading to the formation of spikes. This process is repeated for all directions and neuron pairs in the intermediate layer. As a result, in the output layer is assigned to the central pixel, providing an advanced edge detection capability that provides the dynamic processing of the SNN architecture that mimics the complexities of the Human Visual System. As a result, the differences between the center pixel and neighboring pixels were calculated with the help of values expressed as $Q_1 \dots Q_8$ in the output layer. Thus, the differences between the gray levels of the pixels were obtained in accordance with HVS, not on a gradient basis, and were used for edge detection based on pixel similarity.

Unlike gradient or similarity-based approaches, the proposed method determines the difference calculations according to the firing numbers of neurons with inhibitory and excitatory synaptic connections. Therefore, two different synaptic matrices in the same direction are used to determine the difference information between neighboring pixels. In the model, when one of the M_{5A} and M_{5B} neurons produces spikes at certain intervals, these spikes will be transmitted to the Q_5 neuron in the output layer. In the proposed network structure, all neurons in the intermediate layer have only an excitatory synaptic connection with their corresponding neuron in the output layer. Depending on the number of excitatory spikes from neurons M_{5A} and M_{5B} , the increase in the membrane potential of neuron Q_5 will also cause spike formation at certain intervals. When this process is calculated for all neighboring pixels, 8 different values will be obtained. IF neuron model is used in the proposed method [49]. Additionally, the IF model used was designed according to Eq. 9–15 [31, 45, 50].

The peak conductivity values of the pixels, q_{ex} and q_{ih} in the RF are calculated as:

$$q_{ex} = \alpha G_{xy}; \quad q_{ih} = \beta G_{xy} \quad (9)$$

where G_{xy} value is the gray level value of the pixel in x , y coordinates. α and β are normalization coefficients and are set to 255. The following equations were used for neurons in the intermediate layers.

$$\begin{aligned} \frac{g_{ex}(t)}{dt} &= -\frac{1}{\tau_{ex}}g_{ex}(t) + \frac{w_{ex}q_{ex}}{A_{ex}} \\ \frac{g_{ih}(t)}{dt} &= -\frac{1}{\tau_{ih}}g_{ih}(t) + \frac{w_{ih}q_{ih}}{A_{ih}} \end{aligned} \quad (10)$$

In Eq. 10, τ_{ex} and τ_{ih} are the time constants of excitatory and inhibitory synapses. Synaptic current I_{syn} formed by excitatory and inhibitory connections formulated as;

$$I_{syn} = -g_{ex}E_{ex} - g_{ih}E_{ih} \quad (11)$$

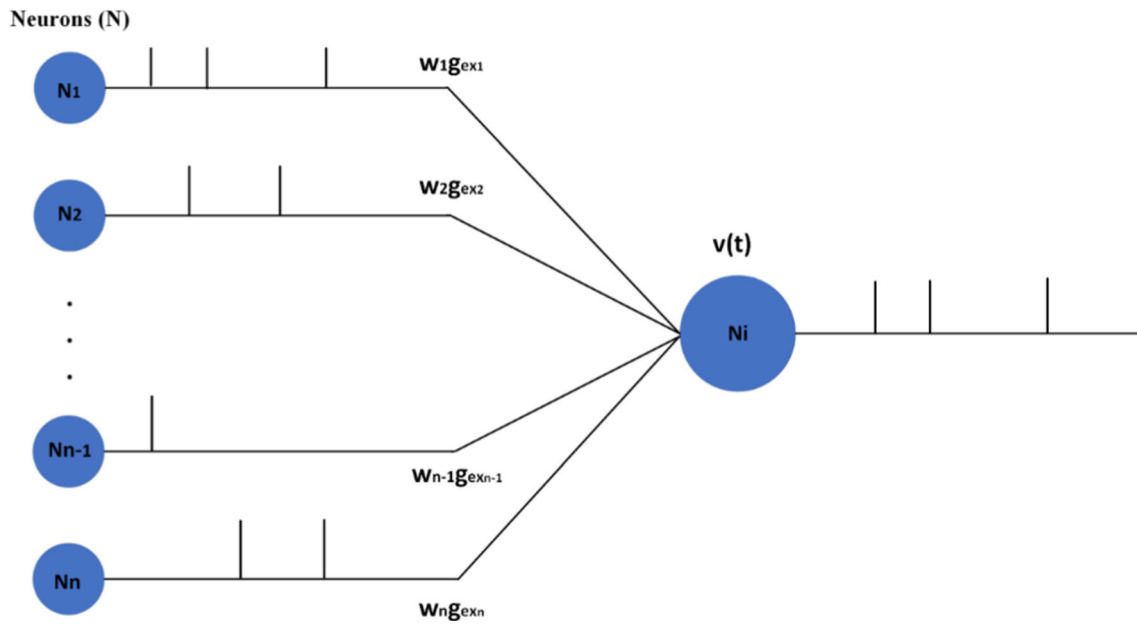


Fig. 6 Synaptic connections of conductance-based IF

Membrane potential of any neuron $v(t)$ is calculated as;

$$v(t) = \frac{1}{g_1} \{ (-e^{\frac{g_1 t}{cm}}) (I_{syn} + 70g_1 + g_1 E_1) + I_{syn} + g_1 E_1 \} \quad (12)$$

The equations given above for any neuron are calculated similarly for 16 neurons. When the membrane potential of neurons reaches the threshold voltage, the neurons form spikes and then return to the initial state. These spikes form separate Spike Trains for each neuron. Spike train belonging to neurons is calculated as S_N

$$S_N = \begin{cases} 1, & \text{if neuron } i \text{ fires a spike at time } t \\ 0, & \text{if there is no spike at time } t \end{cases} \quad (13)$$

8 neurons ($Q_1, \dots, Q_8$) in the output layer are produced for each receptor field. The relevant neurons have synaptic connections with neuron pairs in the intermediate layer, and the output neurons are calculated by the following equations.

$$\begin{aligned} \frac{g_{out}}{dt} &= -\frac{1}{\tau_{out}} g_{out}(t) + \frac{S_{N1}(t) + S_{N2}(t)}{A_{ex}} \\ I_{out} &= -g_{out} E_{out} \\ v_{QN} &= \frac{1}{g_1} \{ (-e^{\frac{g_1 t}{cm}}) (I_{out} + 70g_1 + g_1 E_1) + I_{out} + g_1 E_1 \} \end{aligned} \quad (14)$$

In Eq. 14 g_{out} and τ_{out} represent the time-changeable conductivity and time constant, respectively; I_{out} is the total synaptic current of any Q_N neuron. v_{QN} is the membrane

potential of the Q_N neuron. In the output layer, the firing rates of the spindles produced by the Q_N neuron during the T (ms) period of F_{Q1} per second can be determined as in Eq. 15.

$$F_{Q1} = \frac{10^3}{T} \sum_{t=0}^T S_N(t) \quad (15)$$

4 Image interpolation by using SNN based pixel similarity

In recent interpolation studies, it is seen that the edges are determined as vertical, horizontal and diagonal (45 and 135 degrees). While most of these have detected the edges with linear gradient values, the edges have been detected as non-linear, especially with the SNN-based approach developed by İncetaş recently [28]. In this study, a new approach SNN based similarity transform has put forward for non-linear detection of edges in four different directions.

For example, to determine the horizontal edges, the similarities between the center pixel and its left neighbor and between the center pixel and its right neighbor are calculated in a 3×3 neighborhood. Figure 8 represents a mask used to calculate horizontal edges. Then their average is calculated. For vertical edges, the similarity between the center and the pixel above and the center and the pixel below is calculated and their average is evaluated. Similarly, the average of the similarities between the center pixel and its neighbors is calculated when determining the 45- and 135° edges. Thus, the

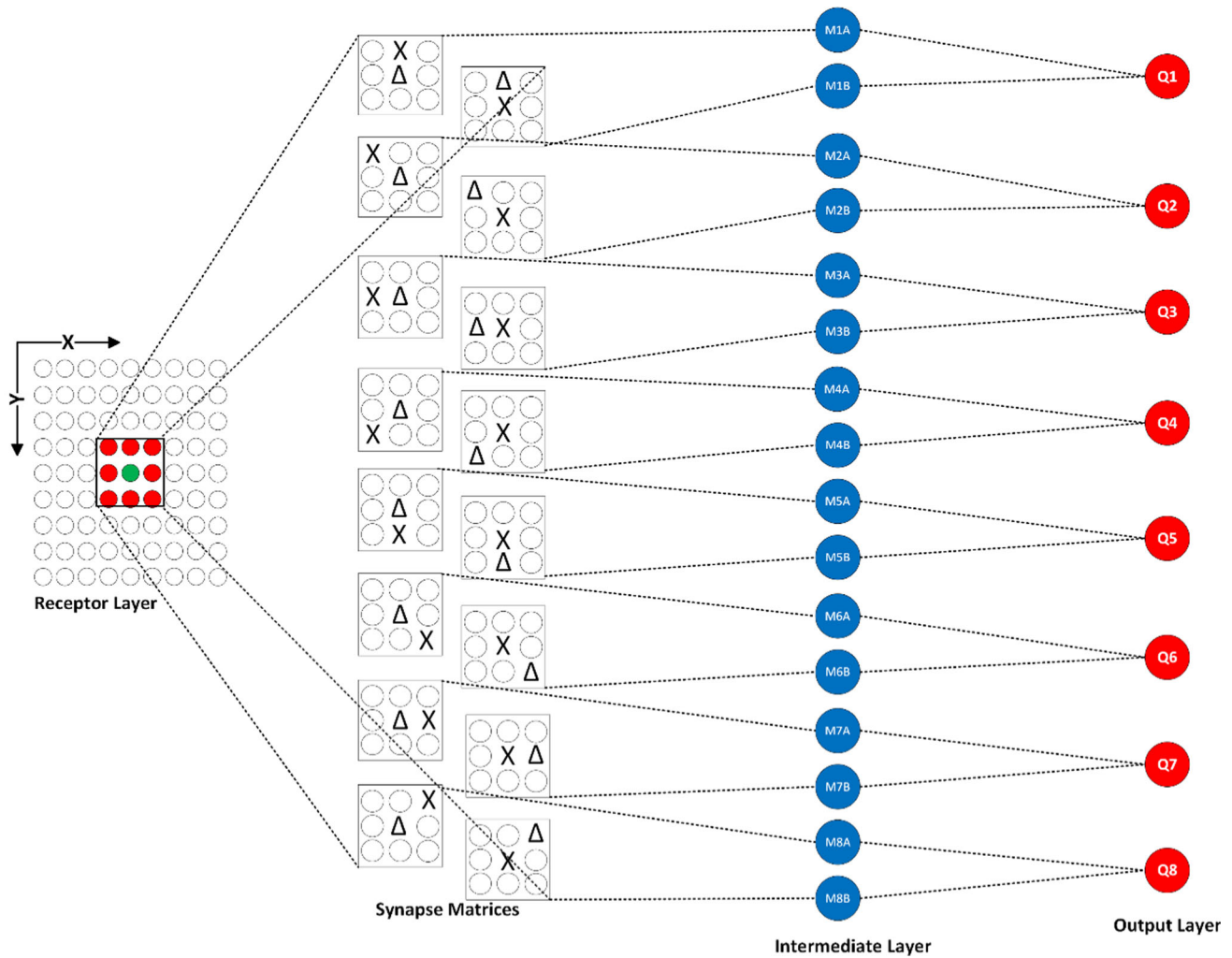


Fig. 7 Proposed SNN based pixel similarity model

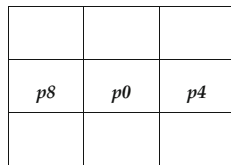


Fig. 8 Mask for horizontal edge detection

edges in 4 different directions are determined as non-linear. In Fig. 9, the edge outputs in different directions can be seen.

Figure 10 represents the block diagram of the proposed interpolation technique. The low-resolution input image I_{LR} is first converted to gray level image. Afterwards, the Similarity Transform approach, which stands out with its non-linear feature, was used for the detection of edge pixels. The related approach makes 4-way edge detection in the image. Because the most important disadvantage in interpolation studies is the information loss in the edge pixels. Therefore, determining the edge pixels and their orientation in low resolution

images is vital in the interpolation process. While determining the edges, the pixels on the diagonal axis are checked first. The difference in pixels on the diagonal, horizontal and vertical axis means that the relevant pixel is an edge pixel. In the diagonal, if there is a difference between two pixels, it means that the corresponding pixel is an edge and the edge direction is assigned as 45° or 135° . This process is also carried out between horizontal and vertical neighborhoods. The edge direction on the diagonal axis is calculated by Eq. 16, while the edge direction on the horizontal and vertical axis is made according to Eq. 17.

$$O(2m, 2n) = \begin{cases} 45^\circ, & S_{45}(m, n) > S_{135}(m, n) \\ 135^\circ, & S_{135}(m, n) > S_{45}(m, n) \end{cases} \quad (16)$$

$$\begin{aligned} O(2m, 2n - 1) &= O(2m - 1, 2n) \\ &= \begin{cases} 0^\circ, & S_0(m, n) > S_{90}(m, n) \\ 90^\circ, & S_{90}(m, n) > S_0(m, n) \end{cases} \end{aligned} \quad (17)$$

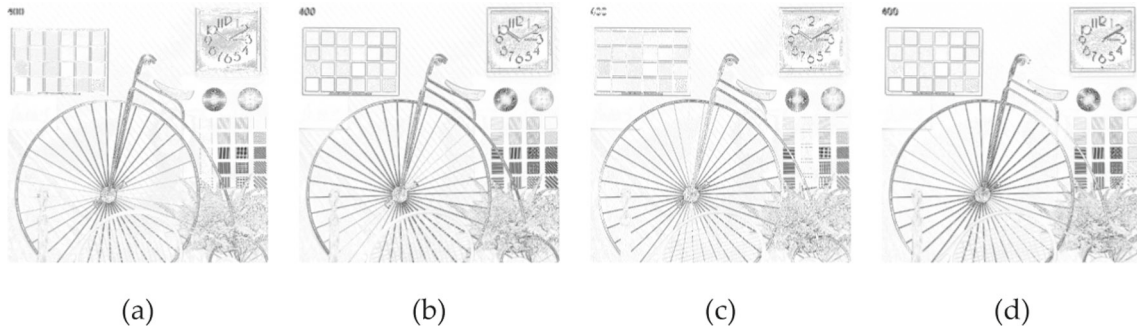


Fig. 9 Edge images in various orientations obtained with proposed: **a** O_0 **b** O_{45} **c** O_{90} **d** O_{135}

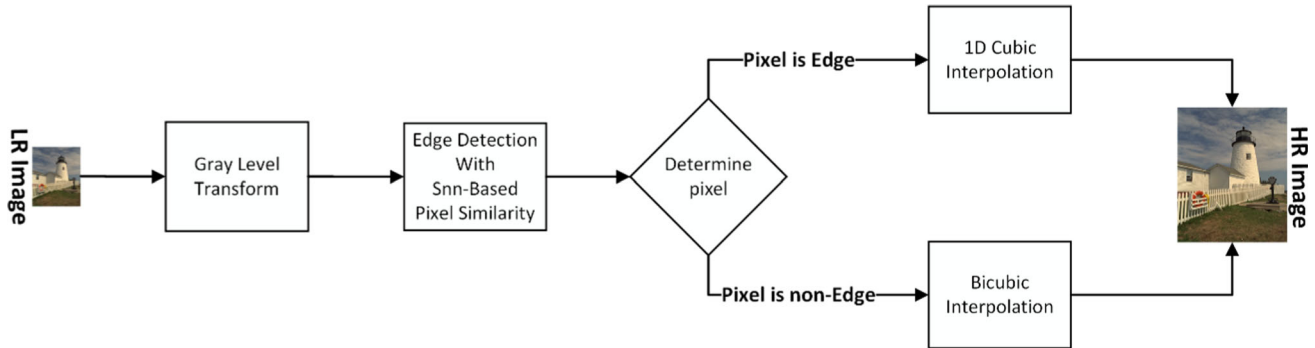


Fig. 10 Block diagram of proposed interpolation approach

Firstly, pixel values at $I_{HR}(2m, 2n)$ location are calculated according to the pixel value at the $I_{LR}(m, n)$ location. Subsequently, $I_{HR}(2m-1, n)$ and $I_{HR}(2m, 2n-1)$ values were determined similarly. Finally, bicubic interpolation processes are applied for each non-edge pixel. For edge pixels, calculations used in many studies were used [26–28, 40]. Related equations are given in Eqs. 18–22.

$$I = \omega(I_x + I_y) + (0.5 - \omega)(I_z + I_t) \quad (18)$$

ω is the interpolation coefficient and its value is set to 0.575. While I expresses the gray level value of any pixel to be interpolated; I_x, I_y, I_z and I_t represent 4 known pixels around the unknown pixel. The corresponding values in the high-resolution I_{HR} image are if $O(r, s) = 45^\circ$;

$$\begin{aligned} I_x &= I_{HR}(r-1, s-1), \quad I_y = I_{HR}(r+1, s+1), \quad I_z \\ &= I_{HR}(r-3, s-3), \quad I_t = I_{HR}(r+3, s+3) \end{aligned} \quad (19)$$

if $O(r, s) = 135^\circ$;

$$\begin{aligned} I_x &= I_{HR}(r+1, s-1), \quad I_y = I_{HR}(r-1, s+1), \quad I_z \\ &= I_{HR}(r+3, s-3), \quad I_t = I_{HR}(r-3, s+3) \end{aligned} \quad (20)$$

if $O(r, s) = 0^\circ$;

$$\begin{aligned} I_x &= I_{HR}(r, s-1), \quad I_y = I_{HR}(r, s+1), \quad I_z \\ &= I_{HR}(r, s-3), \quad I_t = I_{HR}(r, s+3) \end{aligned} \quad (21)$$

if $O(r, s) = 90^\circ$;

$$\begin{aligned} I_x &= I_{HR}(r-1, s), \quad I_y = I_{HR}(r+1, s), \quad I_z \\ &= I_{HR}(r-3, s), \quad I_t = I_{HR}(r+3, s) \end{aligned} \quad (22)$$

5 Experimental results and discussion

The experimental findings were obtained using a computer with an Intel Core i7-12700 K processor and 32 GB of RAM. The average computation time of the proposed approach was calculated using MATLAB soft-ware. To evaluate the performance of the proposed approach, 12 frequently used images were utilized. These images are Bike, Wheel, Boats, Butterfly, House, Cameraman, Baboon, Peppers, Fence, Airplane, Barbara, and Star. In addition, the developed technique was compared and tested against 7 different approaches, namely SNN [28], GEI [27], WTCGI [26], PCI [25], CED [24], CGI [23], and IEDI [22]. Related images were also used in comparison studies. Therefore, the validity and reliability of

Table 1 PSNR evaluation of interpolation approach

	[26]	[27]	[28]	[25]	[29]	[30]	[31]	Proposed
Bike	25.82	25.82	25.9	25.17	25.21	25.85	26.60	27.38
Wheel	21.01	20.98	21.22	20.31	20.57	21.32	21.53	22.53
Boats	29.51	29.56	29.77	29.24	29.32	29.42	29.71	32.05
Butterfly	29.27	29.24	29.31	28.97	28.97	29.26	29.55	30.96
House	32.83	32.71	32.88	32.31	31.87	32.84	33.17	35.38
Cameraman	25.86	25.9	25.81	25.48	25.76	25.83	26.09	27.04
Baboon	22.50	22.41	22.53	22.41	22.35	22.59	22.82	23.54
Peppers	30.88	30.77	30.87	30.47	30.19	30.81	31.12	33.35
Fence	25.7	25.63	25.84	25.61	25.69	25.75	26.01	27.83
Airplane	26.54	26.49	26.59	26.60	26.10	26.61	26.88	28.54
Barbara	23.75	23.64	23.82	23.54	23.41	24.01	24.25	24.93
Stars	34.13	33.94	34.38	33.36	33.71	34.33	34.67	37.45
Average	27.32	27.26	27.41	26.96	26.93	27.39	27.70	29.24

Table 2 SSIM evaluation of interpolation approach

	[26]	[27]	[28]	[25]	[29]	[30]	[31]	Proposed
Bike	0.8808	0.8812	0.8803	0.8751	0.8791	0.8798	0.9071	0.9205
Wheel	0.8621	0.8626	0.8668	0.8644	0.8649	0.8665	0.8986	0.9096
Boats	0.8763	0.8801	0.8794	0.8771	0.8744	0.8796	0.8963	0.9195
Butterfly	0.9721	0.9732	0.9720	0.9718	0.9698	0.9758	0.9992	0.999
House	0.8781	0.8778	0.8789	0.8783	0.8775	0.8780	0.8956	0.8987
Cameraman	0.8711	0.8732	0.8715	0.8704	0.8692	0.8732	0.8976	0.9478
Baboon	0.9125	0.9111	0.9130	0.9121	0.9112	0.9165	0.9403	0.9705
Peppers	0.9032	0.9041	0.9035	0.9029	0.9026	0.9025	0.9278	0.9388
Fence	0.7752	0.7780	0.7785	0.7763	0.7765	0.7723	0.7893	0.8118
Airplane	0.9405	0.9410	0.9401	0.9389	0.9422	0.9412	0.9591	0.9683
Barbara	0.9125	0.9128	0.9130	0.9114	0.9105	0.9118	0.9392	0.9696
Stars	0.9584	0.9603	0.9617	0.9608	0.9610	0.9608	0.9762	0.9975
Average	0.8952	0.8963	0.8966	0.8950	0.8949	0.8965	0.9279	0.9376

the experimental findings are high. In the experiments, as in recent studies, the $1/2 \times 1/2$ sub-sampling of the 12 images was performed. Commonly used methods for sub-sampling include nearest neighbor, linear, and bilinear approaches. However, in this study, a successful method called direct inference was employed [27]. Subsequently, image interpolation was applied to the I_{LR} image obtained through direct inference using a 2×2 upsampling to generate the I_{HR} image. The resulting images were then compared to the original images. PSNR and SSIM metrics were used for the comparison process. Table 1 represents the comparative performance of all methods in terms of PSNR.

When examining the results in Table 1, it is evident that the proposed approach achieves high success. Particularly noteworthy are the interpolation works carried out in recent studies, such as GEI and SNN. On the other hand, when compared to these approaches, the proposed model yields more

effective results. This demonstrates that the critical stage of edge detection, which is a crucial aspect in interpolation works, is performed more successfully with the proposed method. SSIM results, which is another metric, are given in Table 2. The results in Table 2 show compatible with Table 1. The developed technique showed superior performance in all images. In other words, using similarity transform in edge detection has increased the success of interpolation. Output images of GEI, SNN and proposed methods are given visually in Fig. 11. There are 3 methods that give the best results in the related figure. Although it is noteworthy that the methods protect the edge pixels, Table 1 and Table 2 prove the success of the proposed method. In addition, Similarity Based interpolation approach produced faster results than other methods.

When the studies conducted in recent years are examined, it is seen that the methods generally benefit from



Fig. 11 Original and interpolated images: **a** Original **b** [30] **c** [31] **d** Proposed

gradient-based linear structures. However, it is known that non-linear structures represent HVS more successfully. The non-linear structure of the proposed method is one of its important advantages. İncetas [28] model directly used the inter-pixel differences obtained with SNN. In this proposed study, these non-linear differences were used in a similarity-based non-linear edge detection approach. In addition, GEI are successful interpolation techniques developed in 2022. However, it is an important drawback that method needs user dependent parameters. GEI is an interpolation technique that requires calculating regional gradients for each pixel. Noise in the input image negatively affects the gradient estimation, resulting in noise and artifacts as a result of interpolation. Errors made during regional gradient estimation may become more apparent as a result of interpolation and may reduce image quality. The SNN based pixel similarity is completely user-independent and does not need any parameters. CGI, WTCGI and CED approaches often involve complex algorithms, which require high computational cost. Therefore, it is inevitable that processing times will increase, especially for high-resolution images. Additionally, the presence of noise in the input images affects the results of the respective approaches. The performance of contrast-guided interpolation is highly dependent on the quality of the input images. Additionally user dependent. Our proposed method, unlike existing gradient-based linear techniques, determines the difference between pixels non-linearly. Thanks to the proposed SNN model, edges are determined in accordance with HVS. In addition, with the help of the proposed model, calculations are made faster than previously presented SNN models. In addition to all these, it is seen that the success of interpolation is high when compared to state-of-art articles.

In the similarity image, the closer the gray level value of a pixel is to its neighbors, the closer the similarity value is to 255 (or 1). It's attractive easy to interpolate an area with pixels that are similar to each other because the gray level values are also very similar. However, when the entire image is interpolated in the same way, distortions occur, especially at the edges. Techniques developed for this purpose generally include linear structures such as gradients. However, proposed architecture produces results more similar to the non-linear structure of the human visual system. Moreover, the computational cost is competitive compared to various approaches that simulate the human visual system. The calculation time of the methods is shown in Table 3. It is clear from the details of Table 3 that the CED approach performs transactions at the best times. However, the proposed method is clearly superior to methods such as CGI and IEDI.

Although the proposed SNN model and interpolation approach have successful results, the disadvantage of the method is that it does not include a learning process. It will be possible to achieve higher success by using more trained

Table 3 Computation average time (in second) for 12 test images

[26]	[27]	[28]	[25]	[30]	[31]	Proposed
4.15	1.19	1.75	24.06	1.94	2.57	2.47

data sets with learning processes that can be easily included in SNN models.

6 Conclusion

In disciplines such as defense, criminal and medicine, where improving image quality is of vital importance, increasing the size of the image to preserve the edges is very important. In many studies proposed in recent years, the importance of preserving edge information in the interpolation process has been emphasized. For this reason, the studies consist of two basic stages. These stages can be expressed as edge detection and applying the appropriate interpolation technique. The use of linear approaches such as gradients in the detection of edge information is seen in current studies. However, algorithms representing the non-linear feature of the human visual system give more successful results. In this study, a novel nonlinear interpolation method based on pixel similarity with SNN architecture is proposed. The focus of the proposed study is the edge detection, which is the first phase of the interpolation process. In the second stage, the same procedures were carried out as in the current studies. Experimental findings have shown that the proposed method is more effective. In addition, the proposed SNN based pixel similarity approach produced competitive faster results than the others. Numerical evaluations were made using SSIM and PSNR performance criteria. Both numerical results and visual evaluations provided promising interpolation results. Additionally, by including the learning phase, it is planned to future work on edge detection based on pixel similarity and use the determined edges in different image processing areas such as image interpolation, content-based image retrieval and diffusion filter.

Acknowledgements The authors would like to thank to the anonymous reviewers and editor for their insightful comments.

Author contributions MK conceives the idea and writes the draft of the paper, implements the algorithm and performs analysis.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). This article is not supported by any funding.

Data availability All data have appeared in this paper.

Declarations

Conflicts of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Consent for publication All authors appeared in this paper agreed to publication in this journal.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Jiang, C., et al.: Image interpolation model based on packet losing network. *Multimed. Tools Appl.* **79**, 25785–25800 (2020)
- Karwowska, K., Wierzbicki, D.: Using super-resolution algorithms for small satellite imagery: a systematic review. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **15**, 3292–3312 (2022)
- Zheng, J., et al.: Image interpolation with adaptive k-nearest neighbours search and random non-linear regression. *IET Image Proc.* **14**(8), 1539–1548 (2020)
- Guo, A., et al.: An energy-efficient image filtering interpolation algorithm using domain-specific dynamic reconfigurable array processor. *Integration* **96**, 102167 (2024)
- Hung, K.-W., Wang, K., Jiang, J.: Image interpolation using convolutional neural networks with deep recursive residual learning. *Multimed. Tools Appl.* **78**, 22813–22831 (2019)
- Liu, X., et al.: Image interpolation via graph-based Bayesian label propagation. *IEEE Trans. Image Process.* **23**(3), 1084–1096 (2013)
- Baghaie, A., Yu, Z.: Structure tensor based image interpolation method. *AEU Int. J. Electron. Commun.* **69**(2), 515–522 (2015)
- Thévenaz, P., Blu, T., Unser, M.: Interpolation revisited [medical images application]. *IEEE Trans. Med. Imaging* **19**(7), 739–758 (2000)
- Xin, J., et al.: Wavelet-based dual recursive network for image super-resolution. *IEEE Trans. Neural Netw. Learn. Syst.* **33**(2), 707–720 (2020)
- Diana Earshia, V., Sumathi, M.: A guided optimized recursive least square adaptive filtering based multi-variate dense fusion network model for image interpolation. *Signal Image Video Process.* **18**(2), 991–1005 (2024)

11. Kong, L., et al.: FISRCN: a single small-sized image super-resolution convolutional neural network by using edge detection. *Multimed. Tools Appl.* **83**(7), 19609–19627 (2024)
12. Amidror, I.: Scattered data interpolation methods for electronic imaging systems: a survey. *J. Electron. Imaging* **11**(2), 157–176 (2002)
13. Zhang, L., Wu, X.: An edge-guided image interpolation algorithm via directional filtering and data fusion. *IEEE Trans. Image Process.* **15**(8), 2226–2238 (2006)
14. Sánchez-García, E., et al.: A new adaptive image interpolation method to define the shoreline at sub-pixel level. *Remote Sens.* **11**(16), 1880 (2019)
15. Liu, Y., Huang, Q., Sato, K.: Differential filtering algorithm for robot welding seam image enhancement. In: *Journal of Physics: Conference Series*. IOP Publishing (2020)
16. Chao, Y., et al.: Adaptive directional cubic convolution for integrated circuit chip defect image interpolation. *Int. J. Circuits* **15**, 1084–1090 (2021)
17. Gupta, S., Sharma, D.K., Ranta, S.: A new hybrid image enlargement method using singular value decomposition and cubic spline interpolation. *Multimed. Tools Appl.* **81**(3), 4241–4254 (2022)
18. Guan, H., et al.: Using correction parameters to improve real-time video interpolation in low-cost VLSI implementation. *Microelectron. J.* **117**, 105254 (2021)
19. Benseddik, M.L., et al.: Interpolation-based reversible data hiding in the transform domain for fingerprint images. *Multimed. Tools Appl.* **81**(14), 20329–20356 (2022)
20. Deeba, F., et al.: Wavelet integrated residual dictionary training for single image super-resolution. *Multimed. Tools Appl.* **78**, 27683–27701 (2019)
21. Shivagunde, S., Biswas, M.: Single image super-resolution based on modified interpolation method using MLP and DWT. In: *2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI)*. IEEE (2019)
22. Dharejo, F.A., et al.: TWIST-GAN: towards wavelet transform and transferred GAN for spatio-temporal single image super resolution. *ACM Trans. Intell. Syst. Technol. (TIST)* **12**(6), 1–20 (2021)
23. Xue, S., et al.: Wavelet-based residual attention network for image super-resolution. *Neurocomputing* **382**, 116–126 (2020)
24. Sahito, F., et al.: Wavelet-integrated deep networks for single image super-resolution. *Electronics* **8**(5), 553 (2019)
25. Hossain, M.S., et al.: Image resolution enhancement using improved edge directed interpolation algorithm. In: *2019 9th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*. IEEE (2019)
26. Wei, Z., Ma, K.-K.: Contrast-guided image interpolation. *IEEE Trans. Image Process.* **22**(11), 4271–4285 (2013)
27. Ye, W., Ma, K.-K.: Convolutional edge diffusion for fast contrast-guided image interpolation. *IEEE Signal Process. Lett.* **23**(9), 1260–1264 (2016)
28. Zhong, B., Ma, K.-K., Lu, Z.: Predictor-corrector image interpolation. *J. Vis. Commun. Image Represent.* **61**, 50–60 (2019)
29. Zhao, Y., Huang, Q.: Image enhancement of robot welding seam based on wavelet transform and contrast guidance. *Int. J. Innov. Comput. Inf. Control* **18**, 149–159 (2022)
30. Jia, Z., Huang, Q.: Image interpolation with regional gradient estimation. *Appl. Sci.* **12**(15), 7359 (2022)
31. İncetaş, M.O.: Image interpolation based on spiking neural network model. *Appl. Sci.* **13**(4), 2438 (2023)
32. Bhateja, V., Misra, M., Urooj, S.: Non-linear polynomial filters for edge enhancement of mammogram lesions. *Comput. Methods Progr. Biomed.* **129**, 125–134 (2016)
33. Fang, Y., et al.: Stereoscopic image quality assessment by deep convolutional neural network. *J. Vis. Commun. Image Represent.* **58**, 400–406 (2019)
34. Demirci, R.: Similarity relation matrix-based color edge detection. *AEU Int. J. Electron. Commun.* **61**(7), 469–477 (2007)
35. İncetaş, M., et al.: Eşik Seçiminin Benzerliğe Dayalı Kenar Belirlemeye Etkisi (2017)
36. Guvenc, U., Elmas, C., Demirci, R.: Automatic segmentation of color images. *J. Polytech. Politeknik Dergisi* **11**(1), 9–12 (2008)
37. İncetas, M.O., Demirci, R., Yavuzcan, H.G.: Automatic segmentation of color images with transitive closure. *AEU Int. J. Electron. Commun.* **68**(3), 260–269 (2014)
38. Tanyeri, U., İncetaş, M.O., Demirci, R.: Similarity based anisotropic diffusion filter. In: *2016 24th Signal Processing and Communication Application Conference (SIU)*. IEEE (2016)
39. Aydın, M., et al.: Neonatal jaundice detection system. *J. Med. Syst.* **40**, 1–11 (2016)
40. İncetas, M.O., Demirci, R., Yavuzcan, H.G.: Automatic color edge detection with similarity transformation. *Gazi Univ. J. Sci.* **32**(2), 458–469 (2019)
41. Lama, R.K., et al.: Interpolation using wavelet transform and discrete cosine transform for high resolution display. In: *2016 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE (2016)
42. FitzHugh, R.: Mathematical models of excitation and propagation in nerve. *Biol. Eng.* **9**, 1–85 (1969)
43. Nagumo, J., Arimoto, S., Yoshizawa, S.: An active pulse transmission line simulating nerve axon. *Proc. IRE* **50**(10), 2061–2070 (1962)
44. Izhikevich, E.M.: Simple model of spiking neurons. *IEEE Trans. Neural Netw.* **14**(6), 1569–1572 (2003)
45. Wu, Q., et al.: A visual attention model based on hierarchical spiking neural networks. *Neurocomputing* **116**, 3–12 (2013)
46. Kerr, D., et al.: A biologically inspired spiking model of visual processing for image feature detection. *Neurocomputing* **158**, 268–280 (2015)
47. Kerr, D., et al.: Biologically inspired edge detection. In: *2011 11th International Conference on Intelligent Systems Design and Applications*. IEEE (2011)
48. Kerr, D., et al.: Biologically inspired intensity and range image feature extraction. In: *The 2013 International Joint Conference on Neural Networks (IJCNN)*. IEEE (2013)
49. Wu, Q., et al.: Edge detection based on spiking neural network model. In: *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence: Third International Conference on Intelligent Computing, ICIC 2007, Qingdao, China, August 21–24, 2007. Proceedings 3*. Springer (2007)
50. İncetaş, M.O.: Anisotropic diffusion filter based on spiking neural network model. *Arab. J. Sci. Eng.* **47**(8), 9849–9860 (2022)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.