International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3480-3486
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0511

Image Steganography Using Penalty Function Method and PSO

Akhila L[1], Manoj V J[2]
[1]Assistant Professor, Dept. of ECE, Cochin University College of Engineering Kuttanadu, Alappuzha, Kerala, India.
[2] Professor, Dept. of ECE, Cochin University College of Engineering Kuttanadu, Alappuzha, Kerala, India.
Emails: *akhilalmailid@gmail.com[1], manojmvj@gmail.com[2]*

Abstract
This paper presents an innovative methodology in the realm of Pixel Value Differencing with Modulus Function (PVDMF) steganography. It is modelled as an optimization problem to minimize the mean square error between the cover and stego image. A novel approach, Penalty Function Based Particle Swarm Optimization Algorithm (PFBPSOA), is proposed for the optimization. PFBPSOA addresses the optimization problem with constraints by converting it into an unconstrained optimization problem through the application of the penalty function methodology. In this method, for each constraint, a weight term is added to the objective function to prevent constraint violation. Experiment results show that the proposed method preserves good image metrics such as hiding capacity, Peak Signal to Noise Ratio (PSNR), and Quality Index (QI). The proposed methodology is immune to pixel value difference histogram (PDH) analysis.
Keywords: *Steganography, Penalty Function Method, Particle Swarm Optimization, PDH analysis.*

1. Introduction
Steganography is a method of hiding information within other data to prevent detection [1]. Digital images contain repetitive data. They are therefore widely used as carriers in many steganography techniques[1]. When an image is chosen as the carrier, the process is referred to as image steganography [1]. Among the various domains used in steganography, the two most widely discussed methods are the frequency domain and the spatial domain [1]. The frequency domain, also known as the transform domain, applies popular mathematical transforms[1]. In contrast, the spatial domain method, secret data is hidden by directly modifying the pixels of the image. Since no transform or inverse transform is required, this method is faster and less computationally intensive. The simplest spatial domain method in image steganography is the Least Significant Bit (LSB) substitution technique [1]. However, LSB steganography is highly vulnerable to detection, particularly through RS steganalysis [1]. Another spatial domain steganography method is Pixel Value Difference (PVD) steganography [2], which cannot be detected using RS steganalysis. However, this approach suffers from the falling-off-boundary problem [1] and is also vulnerable to Pixel Difference Histogram (PDH) analysis [1]. To address these limitations and improve the quality of the stego image, the Pixel Value Differencing using Modulus Function (PVDMF) method was later introduced [3]. In the PVDMF method, the use of the modulus operation can produce multiple possible stego images, all capable of retaining the hidden data[1]. However, no mechanism is employed to choose the optimal stego image that minimizes the mean square error (MSE) [1]. The PVDMF method is modeled in this paper as an optimization problem[4], with the objective of minimizing MSE while considering the method’s conditions as constraints. Optimization [4] refers to the process of identifying the most effective solution to a problem within the given constraints. Here, Particle Swarm Optimization (PSO)[5] is employed to minimize the MSE between stego and cover images under PVDMF constraints. A new cost function for PSO is formulated by combining the objective function and the constraints of the optimization problem using the penalty function method [1]. As the optimization problem is the minimization of MSE, the new cost function is

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3480-3486
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0511

formulated by adding the penalty functions corresponding to the constraints. If a constraint is not satisfied, a penalty value is added to the objective function in such a way that the cost of the solution is increased. At the end of the execution of PSO algorithm, if the resulting solution fails to meet any of the constraints, the solution is discarded and the algorithm is re-executed. The remainder of this paper is organized as follows: Section II provides a brief overview of PVDMF, Section III introduces Particle Swarm Optimization, and Section IV Mathematical Modeling of Objective function and Constraints in PVDMF, Section V explains the proposed method. Section VI presents the results and discussions, while Section VII concludes the paper.

2. Pixel Value Differencing with Modulus Function (PVDMF)

The literature identifies PVDMF as a robust image steganography technique [3]. A range table is created by dividing the pixel intensity values (0–255) into multiple subranges. For each block, the difference between two adjacent pixels is calculated, and the corresponding subrange is identified. Each subrange has an associated width and determines the number of bits that can be hidden. Based on this, the decimal equivalent of the secret data is obtained. New pixel values are then generated so that the modulus of their sum matches this decimal value. This process is repeated for all blocks in the image, producing the final stego image.

3. Particle Swarm Optimization Algorithm

PSO[5] is a swarm-based meta-heuristic optimization algorithm where the swarms exhibit a collective behavior that makes them move or migrate in a specific direction to a specific place. On the view of optimization, this specific space is the solution space where they find an optimum solution for a specific problem. The flow chart of PSO is shown in Figure 1. Similar to other meta-heuristic optimization algorithms, the initial population is randomly created. The current personal best position of particle is initialized as Pbest. After the evaluation of f itness value, the solution with best fitness value is initialized as global best solution, Gbest. With the help of a velocity parameter, swarm changes the direction and reach in another solution space with other fitness value so that the population is updated. The fitness value of updated solution is evaluated and compared with current personal best solution, Pbest. If it is better than the cost of current personal best solution, the current personal best solution, Pbest is updated. Global best solution, Gbest is finalized by comparing fitness values of updated population with that of Pbest solutions.

4. Mathematical Modeling of Objective function and Constraints in PVDMF

In the proposed method, an image steganography method using PVD with a modulus function, optimized by Penalty function and Particle Swarm Optimization (PSO), is explained. The gray level values in a block are g_0, g_1, g_2 and g_3. We calculate the differences $d_i = g_i - g_0$, i=1,2,3. Each difference falls into one of 13 ranges, which determine how many bits of secret data can be hidden in that difference. The secret bits are converted into decimal values b_i. Then, the pixel values g_0', g_1', g_2', and g_3' are adjusted using PSO. The PSO algorithm searches for the best way to modify these pixels so the image distortion is minimized while the secret data is correctly embedded. To ensure correct embedding, the modified pixels satisfy this condition:

$$(g_0' + g_1') \bmod 2^{t_i} = b_i \text{ , where i=1,2,3}$$

Here, g_0' is the modified smallest pixel in the block, g_i' is the modified pixel difference, t_i is the number of bits embedded, and b_i is the decimal value of the secret data bits. While embedding secret data into the cover image, it is important to ensure that the absolute difference between the modified pixels stays within the allowed range. Additionally, all the newly generated pixel values must stay within the valid grayscale range of 0 to 255. The Mean Square Error (MSE) for each block is calculated as:

$$MSE = \frac{1}{4}\{ (g_0 - g_0')^2 + (g_1 - g_1')^2 + (g_2 - g_2')^2 + (g_3 - g_3')^2 \}$$

Since the image contains M×N/4 such blocks, the

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3480-3486
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0511

overall MSE for the full image is the average of the MSEs of all blocks.

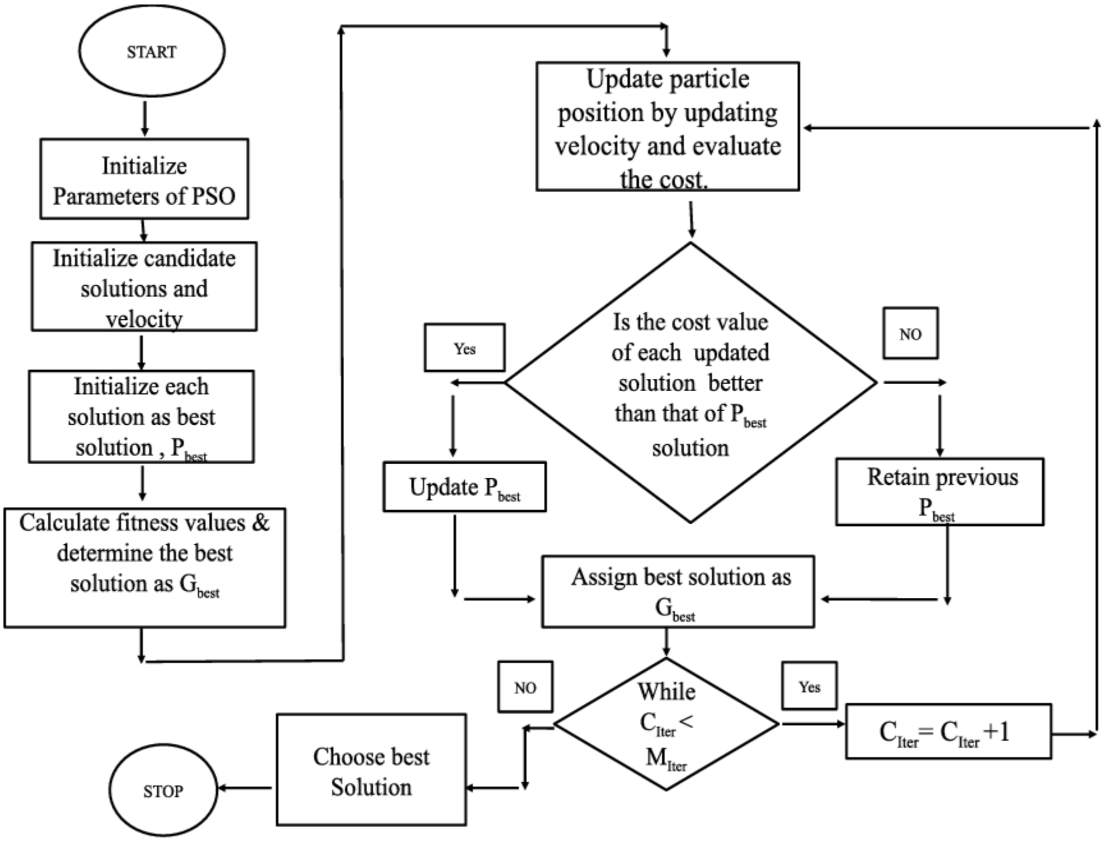

Figure 1 **Flow Chart of PSO**

5. Penalty Function Method (PFM)

The penalty function method [4] is used to solve optimization problems with constraints. It changes a problem with many constraints into one without constraints by adding a penalty term to the objective function. This term handles any violations of the constraint. For minimization problems, the penalty term is added, and for maximization problems, it is subtracted. The original objective function, along with its constraints, is transformed into an unconstrained optimization problem using the Penalty Function Method (PFM). Since the aim is to minimize the Mean Squared Error (MSE), additional weight factors or penalty parameters are incorporated into the objective function. These penalty terms ensure that the constraints are strictly satisfied, which is essential for the exact recovery of the hidden secret data. Considering all the defined penalty terms, the modified cost function can be expressed as:

$$
\begin{aligned}
\mathrm{F}F_{cost} = \frac{1}{4}\{ & (g_0 - g_0{}')^2 + (g_1 - g_1{}')^2 + (g_2 - g_2{}')^2 \\
& + (g_3 - g_3{}')^2 \\
& + \lambda 1 \left((g_0' + g_i') mod 2^{ti} - bi \right) \\
& + \lambda 2 \left(max(0, (g_i' - g_{iUB})) \right) \\
& + \lambda 3 \left(max(0, (g_{iLB} - g'i)) \right) \\
& + \lambda 4 \left(max(0, (g_0' - 255)) \right) \\
& + \lambda 5 \left(max(0, (0 - g_0')) \right) \\
& + \lambda 6 \left(max\left(0, (g_0' \right.\right. \\
& \left.\left. - max(0, G_{min})) \right) \right) \}
\end{aligned}
$$

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3480-3486
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0511

where λ_1, λ_2, λ_3, λ_4, λ_5, and λ_6 are the coefficients of penalty terms in the cost function. g_{iUB} is *min* (255,($g_0{}' + u_s$)) and g_{iLB} is *max* (0, ($g_0{}' + l_s$)). G_{min}= *min*($g_0{}'$, $g_1{}'$, $g_2{}'$). The values of these coefficients are fixed after many trial experiments.

6. Implementation of Proposed Penalty Function-Based PSO Method

The proposed penalty function-based particle swarm optimization (PFBPSO) is also used for the generation of a stego image from a cover image. The proposed Penalty Function-Based Particle Swarm Optimization (PFBPSO) method is employed to generate the stego image from the cover image. Secret data is embedded into each 2×2 block of the cover image. To construct the complete stego image, the PFBPSO process is executed (M×N)/4 times. The detailed steps for generating the new block in the stego image using the PFBPSO method are presented below.

6.1 Step 1 – Initialization of Parameters

The parameters of the PSO algorithm are initialized, and the initial particle population is generated. The penalty function parameters λ_1, λ_2, λ_3, λ_4, λ and λ_6 are assigned appropriate initial values to guide the optimization process.

6.2 Step 2 – Generation of Initial Population

For each block , a population containing P candidate solutions is created. The k^{th} candidate solution in the population, denoted as S_k, consists of four variables: $g'_{0,k}$, $g'_{1,k}$, $g'_{2,k}$, and $g'_{3,k}$.

6.3 Step 3 – Initialization of Velocity

The search process begins by assigning an initial velocity to each particle in the population. The velocity of the k^{th} particle, $v_{i,k}$, is initialized as:

$$V_{ik} = \text{rand}\,(V_{min}, V_{max})$$

where V_{min} and V_{max} denote the lower and upper velocity limits, respectively, and i=0,1,2,3 corresponds to the variable index within the block. The initial velocities help determine the direction and magnitude of each particle's movement during the optimization process.

6.4 Step 4 – Cost Value Evaluation

The cost of each candidate solution in the initial population is computed using FF_{cost}. This evaluation determines the fitness of each solution with respect to the optimization objective.

6.5 Step 5 – Initialization of P_{best}

Each particle in the population is associated with its personal best solution, P_{best}. The position and cost of P_{best} are initialized with the position vector and corresponding cost from the initial population. For the k^{th} particle in a block , the personal best position and its cost are represented as P_{best} and $CP_{best,k}$. respectively.

6.6 Step 6 – Initialization of G_{best}t

The global best solution, G_{best}, is initialized by selecting the candidate solution from the initial population that has the minimum cost value.

6.7 Step 7 – Velocity Update

The velocity of each particle is updated using the following equation:

$$Vel_k = w * Vel_k + C_1 * R_1(P_{best,k} - S_k) + C_2 * R_2(G_{best} - S_k)$$

where R_1 and R_2 are uniformly distributed random numbers in the range [0,1], w is the inertia weight, and C_1 and C_2 are the cognitive and social acceleration coefficients, respectively.

6.8 Step 8 – Position Update

The position of each particle is updated based on its velocity:

$$S_k = S_k * Vel_k$$

This update allows particles to explore the search space while balancing exploitation of known good solutions and exploration of new regions.

6.9 Step 9: Updating P_{best}

In this step, each particle's personal best position (P_{best}) is updated. For each particle, compare the cost of its current position in the updated population with the cost at its stored P_{best} position. If the current position has a lower cost (i.e., better solution), update the P_{best} position with the current position.

6.10 Step 10: Updating G_{best}

In this step, the global best position (G_{best}) is

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3480-3486
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0511

updated. Compare the cost of the best solution found in the updated population with the cost at the current G_{best} position. If the new best solution has a lower cost, update G_{best} to this new best position.

6.11 Step 11: Termination

The algorithm terminates when the maximum number of iterations is reached. At this point, the current best solution is considered the optimal solution if all problem constraints are satisfied. If the constraints are not satisfied, the solution is discarded, and the algorithm is restarted (re-executed) until an optimal solution meeting all constraints is found. Table 1 shows Image Metric Performance of PFBPSO Figure 2 shows Cover Images Figure 3 shows Stego Images

Table 1 **Image Metric Performance of PFBPSO**

Cover Image	Hiding Capacity	PSNR	QI
Lenna	5,70,362	37.99	0.9956
Baboon	7,23,757	31.24	0.9811
Peppers	5,88,961	37.22	0.9964
Tank	6,12,964	37.85	0.9898
Airplane	5,85,401	40.22	0.9895
Truck	5,77,224	37.85	0.9898
Elaine	4,16,909	37.44	0.9925
Couple	5,76,218	36.75	0.9903

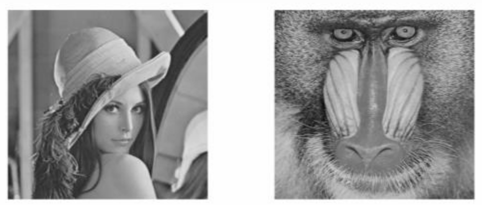

Figure 2 **Cover Images**

Figure 3 **Stego Images**

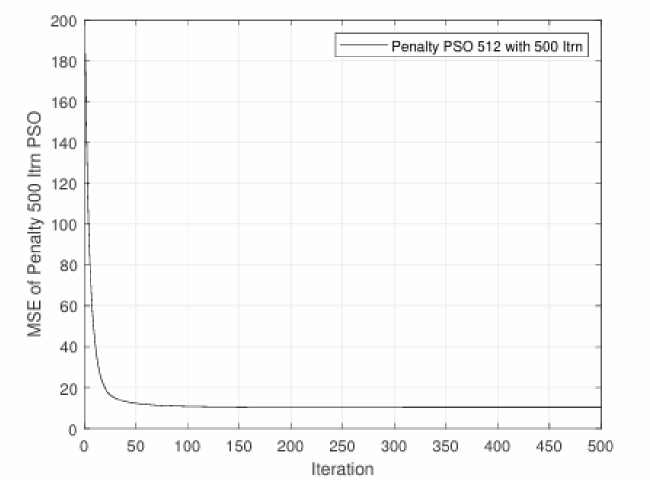

Figure 4 **Evolution Processes Of PFBPSO(Lena)**

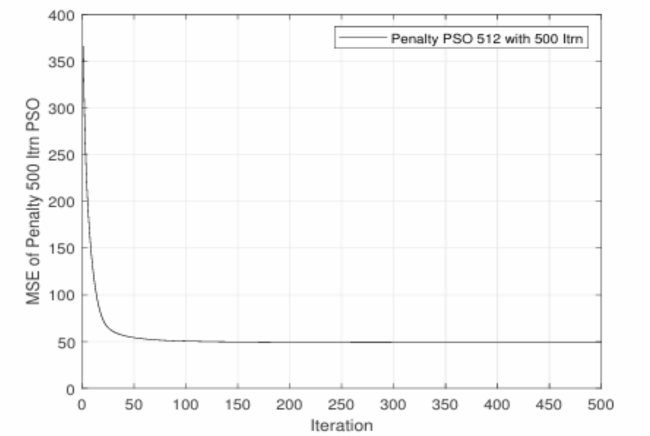

Figure 5 **Evolution Processes of PFBPSO(Baboon)**

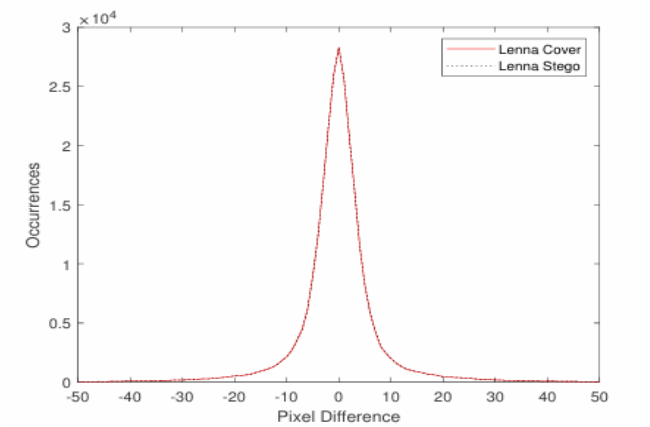

Figure 6 **PDH curve of Lena**

International Research Journal on Advanced Engineering Hub (IRJAEH)
e ISSN: 2584-2137
Vol. 03 Issue: 09 September 2025
Page No: 3480-3486
https://irjaeh.com
https://doi.org/10.47392/IRJAEH.2025.0511

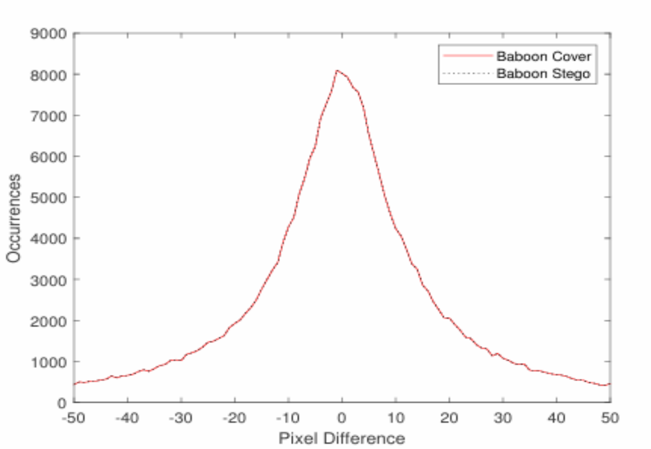

Figure 7 PDH curve of Baboon

7. Results And Discussion

7.1 Results

The performance of the proposed PFBPSO algorithm for digital image steganography was implemented and evaluated using MATLAB R2019b. A secret dataset consisting of a randomly generated binary string was embedded within cover images. The cover images used were grayscale bit map (BMP) files with 8-bit depth and a resolution of 512 × 512 pixels. These images were sourced from the University of Southern California–Signal and Image Processing Institute (USC SIPI) image database [6]. Since the embedding process selects one 1 × 4 block at a time, the number of variables, N, is fixed at four. The PSO parameters—namely, inertia weight (w), cognitive coefficient (C1), social coefficient (C2), and velocity scaling factor—were empirically set to 0.729, 1.494, 1.494, and 0.1, respectively, after evaluating the algorithm's performance across various parameter combinations. The population size was fixed at 50. The penalty function-based method combined with PSO aims to minimize the cost. The penalty parameters λ1, λ2, λ3, λ4, λ5 and λ6 were determined through extensive experimentation. Specifically, λ1 is set to 300, while λ2 and λ3 are fixed at 100. The parameters λ4, λ5, and λ6 are all assigned a value of 50. In this approach, the constraints are checked for each 2 × 2 block. If a solution does not satisfy the constraints, it is discarded, and the algorithm is re-executed until an optimal solution that meets all constraints is obtained.

7.2 Discussion

The result analysis shows that the proposed method is good in terms of PSNR and hiding capacity. The average evolution processes are graphically represented for stego images of Lena and Baboon. MSE is plotted on the Y axis, and the number of iterations is plotted on the X axis. Table 1 shows that QI values are nearly equal to one in this method. It is observed that PDH plots of cover and stego images overlap in images, and steganography is insensible for the proposed method.

Conclusion

A new steganographic method is proposed by modeling data encryption using the PVDMF method as an optimization problem. The error introduced due to pixel modifications in each 2 × 2 block is evaluated using the Mean Squared Error (MSE). In PFBPSO, the cost function is formulated by adding penalty functions corresponding to the constraints directly into the objective function. From the result analysis, it is clear that the proposed method is better than various existing methods in terms of hiding capacity, PSNR, and QI. The method resists the steganalysis technique PDH analysis.

Acknowledgements

No financial support was received from any source for the work being published.

References

[1]. Aditya Kumar Sahu, Monalisa Sahu, "Digital image steganography and steganalysis: A journey of the past three decades," De Gruyter, Open Computer Science, October 2020, 10, pp. 296–342
[2]. Wu PC, Tsai WH. A Steganographic Method for Images by Pixel-value Differencing, Pattern Recognition Letters 2003;1613-1626.
[3]. C.-M. Wang, N.-I. Wu, C.-S. Tsai, M.-S. Hwang, A high quality steganographic method with pixel-value differencing and modulus function, The Journal of Systems and Software, 81, pp.150-158, 2008.

[4]. Singiresu S. Rao, Engineering Optimization: Theory and Practice.
[5]. Kennedy, J.; Eberhart, R. (1995), Particle Swarm Optimization. Proceedings of IEEE International Conference on Neural Networks. Vol. IV. pp. 1942–1948. doi:10.1109/ICNN.1995.488968.
[6]. The USC-SIPI Image Database, http://sipi.usc.edu.The CV online Database, https://homepages.inf.ed.ac.uk.