Modelling and estimate of the LTWM noise parameters

Bojan Prlinčević¹, Zoran Milivojević²

¹Higher Technical Professional School Zvecan, St. Nusiceva 6, Zvecan, Serbia
²The Academy of Applied Technical Studies, St. Aleksandra Medvedeva 20, 18000 Nis, Serbia

b.prlincevic@vts-zvecan.edu.rs, zoran.milivojevic@vtsnis.edu.rs

Abstract. In the first part of this paper, a statistical analysis of the noise in the image, as a result of the insertion of the watermark by Liu and Tan algorithm, was performed. After that, using the base of test image the noise model was created. Based on the model, an algorithm for estimating the statistical parameters of noise, LTWM algorithm, ($\mu$ and $\sigma^2$) in the image with a watermark, was created. In the second part of this paper there is a description of the experiment in which a comparative analysis of the accuracy of estimating statistical parameters by the proposed LTWM algorithm and Algorithms from [3], [6] - [8], was performed. Algorithms have been applied on watermarked images from the NRCS base. The results of MAE and MSE between true and estimated variance are shown in tables and diagrams. In addition, the execution time of the tested algorithms was measured. A detailed comparative analysis has shown that the proposed LTWM algorithm was more precise, and has a shorter execution time, compared to the analysed algorithms.

1. Introduction

The efficiency of algorithms for digital image processing (edge detection, resizing, translation, rotation and geometric correction, scaling, improve image quality, image compression, image watermarking, visual pattern recognition, face recognition, object classification,...) [1], [2] can be highly degraded due to noise presence in the image. The degree of degradation of efficiency of the algorithms is directly proportional to the intensity of the noises in the image. In most cases, the noise can be modeled by Gaussian distribution and presented by statistical parameters, the mean $\mu$ and the variance $\sigma^2$ (e.g. the amplifier noise, the shot noise, the grain noise of photographic film...) [3], [4]. In the image denoising literature, noise is often assumed to be zero-mean additive white Gaussian noise (AWGN) [3], [5]. The first step in denoising is the estimation of noise parameters (e.g. Algorithm from [6], [7], Algorithm from [8], Algorithm from [3], and many others [9] - [11]).

Inserting a digital watermark leads to degradation of the visual characteristics of the image. The degree of degradation is directly dependent on the insertion factor. Decreasing the insertion factor leads to a decreasing of visual image degradation. However, the problem of reliable watermark extraction is related with the watermark low insertion factor. The level of image noise depends of the inserted watermark structure, i.e. of the percentage of white pixels in the watermark. Many transformations are used to insert the watermark into the image: a) Discrete Cosine Transform (DCT) [12], b) Discrete Wavelet Transform (DWT) [13], c) Singular Value Decomposition (SVD) [14], d) Schur Decomposition, (SD) [15], and others. A commonly used algorithm for inserting a watermark is the Liu and Tan (LT) algorithm, based on the SVD transformation [16].
The authors of this paper tried to answer the questions: a) whether the noise caused from the insertion of a watermark by LT algorithm could be modeled? and b) whether prediction of noise parameters can be made based on the noise model? The answers to the asked questions were obtained through realization of the experiments. Based on the results of the experiments, it was shown that the images noise caused by inserted watermark, inserted with the LT algorithm (LTWM noise), can be modeled as Gaussian noise. Based on that, in order to predict the statistical parameters (μ, σ²), an noise (LTWM) model was created. The LTWM model was created based on statistical parameters of standard test images from the base I. The accuracy of the estimation of the noise parameters was made by a comparative analysis with the estimation results of: a) the Algorithm from [6], [7], b) the Algorithm from [8] and c) the Algorithm from [3]. The mean absolute error (MAE) and mean square error (MSE) was used as a measure of comparing the estimation accuracy. In addition, a comparative analysis of the execution time (Tₑ) of algorithms was performed. In the experiment images from the NRCS base (https://photogallery.sc.egov.usda.gov/photogallery/#/) [17], [18] was used.

The paper is organized as follows. In the section 2, analysis of the noise parameters has been performed. Section 3 shows the LTWM model creation proceeding. Section 4 describes the algorithm for estimating the parameters of the LTWM noise. The section 5 describes the experimental results and analysis of the results. Section 6 is a conclusion.

2. The algorithm for estimation LTWM noise statistical parameters
In this chapter an algorithm for estimating the statistical parameters of the noise, which is effect of watermark insertion with LT algorithm, has been described. Confirmation that the LTWM noise can be modeled by Gaussian distribution, was obtained by comparing the true histogram and the histogram obtained based on the estimated parameters μ and σ². The algorithm for estimating the statistical parameters of the LTWM noise, is carried out as follows:

**Input:** Host image X, binary watermark W, image dimensions M × N, insertion factor α, percentage p of white pixels in the watermark.

**Output:** μ, σ².

**Step 1:** SVD decomposition of the image X:

\[ X = U \cdot D \cdot V' \] (1)

where \( U \) is the left orthogonal matrix (\( M \times M \)), \( V \) is the right orthogonal matrix (\( N \times N \)), and \( D \) is the diagonal singular matrix (\( M \times N \)).

**Step 2:** Generating the watermark W, with the p percentage of the white pixels spatially uniformly distributed.

**Step 3:** Inserting the watermark W into the matrix D, with the insertion factor α:

\[ D' = D + \alpha \cdot W , \] (2)

**Step 4:** Creating the watermarked image:

\[ X_W = U \cdot D' \cdot V' , \] (3)

**Step 5:** The LTWM noise matrix is:

\[ N_{XW} = X - X_W , \] (4)

**Step 6:** Statistical parameters of the \( N_{XW} \) noise are:

\[ \mu = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} N_{XW} (i, j) , \] (5)

and
\[ \sigma^2 = \frac{1}{MN-1} \sum_{i=1}^{M} \sum_{j=1}^{N} (N_{XW}(i,j) - \mu)^2. \]  

**Step 7:** Based on the estimated statistical parameters \((\mu, \sigma^2)\), the Gaussian distribution is modeled:

\[ f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \]

Applying the described algorithm to the test images from base \(I\), the results are obtained and presented by histograms. In Fig. 1 are shown images: a) Lena (Fig. 1.a), b) watermark \(W\) (Fig. 1.b), and c) histograms (Fig. 1.c) (the distribution of the noise pixel intensity (4) (marker: '○'), and calculated Gaussian distribution (7) (solid line) for the test image Lena). Used watermark (Fig. 1.b) was generated with \(p = 50\%\) white spatially uniformly distributed pixels. Presented results are for inserted watermark \(W\), with the insertion factor \(\alpha = 25\).

![Figure 1. The image Lena (a), Watermark (b), and distribution of the pixel intensity of the LTWM noise and the Gaussian distribution for presented test image (c).](image)

### 3. Creating of the LTWM model

In this section describes the algorithm for creating the LTWM model. The LTWM model has been created on the basis of the results obtained by Section 2. The algorithm is carried out as follows:

**Input:** Image base \(I\), with \(B\) images, image dimensions \(M \times N\), insertion factor \(\alpha_{\text{min}}, \Delta\alpha, \alpha_{\text{max}}\), percentage of white pixels \(p_{\text{min}}, \Delta p, p_{\text{max}}\), luminance \(Y_{r_{\text{min}}}, \Delta Y_r, Y_{r_{\text{max}}}\).

**Output:** LTWM model \(X_m\).

**FOR** \(y_r = Y_{r_{\text{min}}}: \Delta Y_r: Y_{r_{\text{max}}}\)

**FOR** \(i = 1: B\)

**Step 1:** Loading the image \(X_i\) from the base \(I\).

**Step 2:** Determining the luminance of the image \(X_i\):

\[ y_i = \frac{1}{M \cdot N} \sum_{i=1}^{M} \sum_{j=1}^{N} X_{i,j}, \]  

**Step 3:** Generating the image \(X\) with the luminance \(y_r\)

\[ X = k \cdot X_i, \]  

where \(k = y_r / y_i\).

**FOR** \(\alpha = \alpha_{\text{min}}: \Delta\alpha: \alpha_{\text{max}}\)

**FOR** \(p = p_{\text{min}}: \Delta p: p_{\text{max}}\)

**Step 4:** Generating the binary watermark \(W\) with percentage of white pixels \(p\);
Step 5: Creating an watermarked image \( X_W \) with the insertion factor \( \alpha \) (equations (1) - (3));

Step 6: LTWM noise is \( N_{XW} = X - X_W \), equation (4), with statistical parameters \( \mu \), equation (5), and \( \sigma^2 \), equation (6), \( \implies Q(\alpha, p, i, y_r) = \mu \), and \( S(\alpha, p, i, y_r) = \sigma^2 \).

END \( \alpha \)

END \( \mu \)

END \( \sigma \)

Step 7: Generating LTWM model \( X_m = \{ X_{m\mu}, X_{m\sigma^2}\} \), where \( X_{m\mu} = \frac{X_{mp}}{B} \) and \( X_{m\sigma^2} = \frac{X_{ms^2}}{B} \) are components of the model.

Components of the LTWM model \( X_m \), for the image with \( Y = 30 \), are shown in Fig. 2.a (component \( X_{m\mu} \)) and Fig. 2.b (component \( X_{m\sigma^2} \)). The parameters of the algorithm for creating LTWM model (Section 3) are \( \alpha_{\min} = 1 \), \( \Delta \alpha = 1 \), \( \alpha_{\max} = 50 \), \( p_{\min} = 1 \), \( \Delta p = 5 \), \( p_{\max} = 100 \), \( Y_{r,\min} = 10 \), \( \Delta Y_r = 10 \) and \( Y_{r,\max} = 250 \).

4. Estimation of the LTWM noise

The section describes algorithm for estimating of the LTWM noise parameters of the image \( X \), for mean illumination \( Y \), to which the binary watermark \( W \) is inserted. The algorithm for estimation LTWM noise (LTWM algorithm) consists of the following steps:

Input: Test image \( X \), binary watermark \( W \), LTWM model \( X_m = \{ X_{m\mu}, X_{m\sigma^2}\} \), insertion factor \( \alpha \).

Output: LTWM noise statistic parameters ( \( \mu_e \), \( \sigma^2_e \)).

Step 1: Luminance of the image \( X \) (without a watermark) is \( Y = \overline{X} \).

Step 2: Percentage \( p \) of the white pixels in the watermark \( W \) is:

\[
p = \left( \frac{N_W}{MN} \right) \cdot 100 \, [\%],
\]

where \( N_W \) – is the number of white pixels in the watermark.

Step 3: Estimation of the LTWM noise parameters by using the components of the LTWM model:

\[
\mu_e \leftarrow X_{mp}[p, \alpha, Y], \quad \sigma^2_e \leftarrow X_{ms^2}[p, \alpha, Y].
\]

5. Experimental results and analysis

This accuracy of the statistical parameters estimation of the LTWM noise by using the LTWM model (Section 4) was determined by the following experiment.

5.1. Experiment
The experiment was carried out, within which the estimation of the statistical parameters of the LTWM noise $p, \sigma^2$ was performed using the algorithm for comparative analysis. The estimation was performed using the LTWM algorithm (Section 4) on the LTWM model (Section 3). The noise matrix $N_{xyw}$, equation (4), was created from the watermarked image $X_w$, and the statistical parameters were calculated, equation (5) and equation (6), which were considered as the accurate values in the further analysis. After that, the accuracy of the parameter estimation was tested and compared with the estimation results of the noise parameters in the watermarked image $X_w$, using the algorithms shown in [6], [7], [8] and [3]. As a measure of quality, the the mean absolute error MAE and MSE were used. In addition to the accuracy of the estimation, the analysis of the time $T_o$ of execution of the tested algorithms was performed. The algorithms were performed in Matlab on the personal computer: processor Intel(R) Core(TM) i5 - 3230M CPU @ 2.6 GHz, RAM memory 8 GB, 64-bit OS.

5.2. Comparative analysis

Comparative analysis of the estimation accuracy of the statistical parameters of the LTWM noise consists of the following steps:

**Input:** Image base $P = \{X_1, X_2, \ldots, X_B\}$, number of tested images $B$, LTWM model $X_m$, insertion factor limits $a_{\text{min}}, a_{\text{max}}$, iterative step $\Delta a$, percentage of white pixels in the watermark $p$.

**Output:** Estimation errors of LTWM $\text{MAE}_X$, Algorithm from [6], [7] $\text{MSE}_P$, Algorithm from [8] $\text{MSE}_G$, and Algorithm from [3] $\text{MSE}_W$ algorithms.

**FOR** $i = 1 : B$

**Step 1:** Selecting the image $X_i$ from the image base $P$: $X_i \leftarrow P = \{X_1, X_2, \ldots, X_i, \ldots, X_B\}$, and determining the luminance $Y$, equation (8), of selected image.

**Step 2:** Generating the binary watermark $W_p$ with $p \%$ of white pixels uniformally spatially distributed, FOR $\alpha = a_{\text{min}} : \Delta a : a_{\text{max}}$.

**Step 3:** Creating an watermarked image $X_w$ by inserting the binary $W_p$ into the image $X_i$ with the insertion factor $\alpha$, by using the LT algorithm, equations (1) - (3): $X_w \leftarrow LT(X_i, W_p, \alpha)$.

**Step 4:** Determining variance $\sigma^2_{\text{est}_i}$, equation (6), (true value), noise matrix $N_{xyw}$, equation (4).

**Step 5:** Estimation variance of the LTWM noise using:

a) LTWM algorithm (Section 4) (LTWM model $X_m$, component $X_{m_{\alpha}}$);

\[ \sigma^2_{\text{est}_{\alpha}}(\alpha) \leftarrow X_{m_{\alpha}}(p, \alpha, Y), \]

b) Algorithm from [6], [7]:

\[ \sigma^2_{\text{est}_f}(\alpha) \leftarrow \text{Algorithm}[6],[7](X_w), \]

c) Algorithm from [8]:

\[ \sigma^2_{\text{est}_G}(\alpha) \leftarrow \text{Algorithm}[8](X_w), \]

d) Algorithm from [3]:

\[ \sigma^2_{\text{est}_W}(\alpha) \leftarrow \text{Algorithm}[3](X_w). \]

**Step 6:** Absolute error of estimated variance $\Delta \sigma^2_{\text{al}_i}(\alpha) = \left| \sigma^2_{\text{est}_i}(\alpha) - \sigma^2_{\text{al}_i}(\alpha) \right| \Rightarrow a) \Delta \sigma^2_{\text{LT}_i}, \ b) \Delta \sigma^2_{\text{LT}_f}, \ c) \Delta \sigma^2_{\text{LT}_G}, \ \text{and} \ d) \Delta \sigma^2_{\text{LT}_w} \Rightarrow \text{END} \ a$.

**Step 7:** Sum of absolute error of estimated variances for all images $\Sigma \Delta \sigma^2_{\text{al}_i}(i) = \Sigma \Delta \sigma^2_{\text{al}_f} + \Sigma \Delta \sigma^2_{\text{al}_G} \Rightarrow a) \Sigma \Delta \sigma^2_{\text{LT}_f}, \ b) \Sigma \Delta \sigma^2_{\text{LT}_G}, \ c) \Sigma \Delta \sigma^2_{\text{LT}_G}, \ \text{and} \ d) \Sigma \Delta \sigma^2_{\text{LT}_w}$.

**Step 8:** Sum of estimated variances $\Sigma \sigma^2_{\text{al}_i}(i) = \Sigma \sigma^2_{\text{al}_f} + \Sigma \sigma^2_{\text{al}_G} \Rightarrow a) \Sigma \sigma^2_{\text{LT}_f}, \ b) \Sigma \sigma^2_{\text{LT}_G}, \ c) \Sigma \sigma^2_{\text{LT}_G}, \ \text{and} \ d) \Sigma \sigma^2_{\text{LT}_w}$.

**END** $i$

**Step 9:** Mean absolute error (MAE) of estimated variance $\text{MAE}_{\text{al}_i} = \Sigma \Delta \sigma^2_{\text{al}_i} / B \Rightarrow a) \text{MAE}_{\text{LT}_f}, \ b) \text{MAE}_{\text{LT}_G}, \ c) \text{MAE}_{\text{LT}_G}, \ \text{and} \ d) \text{MAE}_{\text{LT}_w}$.  

5
Step 10: Mean of estimated variances
\[ \overline{\sigma^2_{alg}} = \frac{1}{B} \sum_{i=1}^{B} \sigma^2_{alg} \]

\[ \Rightarrow \text{a) } \overline{\sigma^2_{LT}}, \text{ b) } \overline{\sigma^2_{T}}, \text{ c) } \overline{\sigma^2_{G}}, \text{ and d) } \overline{\sigma^2_{W}}. \]

Step 11: MSE of estimated variance
\[ \text{MSE}_{alg} = \frac{1}{\alpha_{max} - \alpha_{min}} \sum_{\alpha = \alpha_{min}}^{\alpha_{max}} \left( \sigma^2_{\tau}(\alpha) - \sigma^2_{alg}(\alpha) \right)^2 \]

\[ \Rightarrow \text{a) } \text{MSE}_{LT}, \text{ b) } \text{MSE}_{T}, \text{ c) } \text{MSE}_{G}, \text{ and d) } \text{MSE}_{W}. \]

5.3. Experimental Base
For purpose of the realization the experiments, image bases were created:

a) the image base of the standard test images \( I = \{ \text{Lena, Girl, Baboon, Peppers, Barbara, Boat} \} \), used to create the LTWM model \( X_m \),

b) the image base \( P \), formed from 100 images from NRCS base [17], [18],

c) binary watermark with \( p \) [%] white, uniformly distributed pixels.

5.4. Results
The following results were obtained by applying the algorithm for comparative analysis (Section 5) with the parameters: \( \alpha_{min} = 1 \), \( \alpha_{max} = 50 \), \( \Delta \alpha = 1 \), \( p = 50\% \) on the image base \( P = \{ X_1, X_2, ..., X_B \} \), \( B = 100 \) (images from NRCS base \( P \)), and by using LTWM model \( X_m \). On the figures are shown results of applied measures, which represents mean value for \( B = 100 \) images.

Fig. 3.a shows the estimated variances \( \overline{\sigma^2_{LT}} \) of LTWM noise for: a) True value (marker: ‘●’) (4), b) LTWM (marker: ‘■’), c) Algorithm from [6], [7] (marker: ‘▼’), d) Algorithm from [8] (marker: ‘▲’), and e) Algorithm from [3] (marker: ‘♦’), for \( \alpha = \alpha_{min} : \Delta \alpha : \alpha_{max} \). Fig. 3.b shows a part of the diagram from Fig. 3.a, for \( \alpha = 0 - 10 \), where LTWM noise are not visible in the watermarked image.

Fig. 4.a shows the mean absolute errors MAE of estimated variances: a) LTWM algorithm \( \text{MAE}_{LT} \) (marker: ‘●’), b) Algorithm from [6], [7] \( \text{MAE}_T \) (marker: ‘▲’), c) Algorithm from [8] \( \text{MAE}_G \) (marker: ‘■’) and d) Algorithm from [3] \( \text{MAE}_W \) (marker: ‘♦’)) for \( \alpha = \alpha_{min} : \Delta \alpha : \alpha_{max} \). Fig. 4.b shows a part of the diagram from Fig. 4.a, for \( \alpha = 0 - 10 \), where LTWM noise are not visible in the watermarked image.

Table 1. shows mean square error MSE of estimated variance, for parts of insertion factor: a) \( \alpha = 0 - 50 \) (MSE\(_{0-50}\)), b) \( \alpha = 0 - 10 \) (MSE\(_{0-10}\)) where noise is not visible in the watermarked image, and c) \( \alpha = 11 - 50 \) (MSE\(_{11-50}\)) where noise is visible in the watermarked image. Table 2. shows the execution time for: a) LTWM \( (T_{LTWM}) \), b) Algorithm from [8] \( (T_G) \), c) Algorithm from [6], [7] \( (T_T) \), and d) Algorithm from [3] \( (T_W) \) algorithms.

| Algorithm              | MSE\(_{0-30}\) | MSE\(_{0-10}\) | MSE\(_{11-30}\) |
|------------------------|----------------|----------------|----------------|
| LTWM                   | 804.94         | 0.4303         | 1026.2         |
| Algorithm from [6], [7]| 20931          | 2.4574         | 26687          |
| Algorithm from [8]     | 24319          | 1.4384         | 31006          |
| Algorithm from [3]     | 24837          | 1210.8         | 31335          |

| Algorithm              | \( T_e \) (s)  |
|------------------------|----------------|
| LTWM                   | \( T_{e,LTWM} = 0.0176 \) |
| Algorithm from [8]     | \( T_{e,G} = 0.0260 \) |
| Algorithm from [6], [7]| \( T_{e,T} = 0.3831 \) |
| Algorithm from [3]     | \( T_{e,W} = 0.61285 \) |
5.5. Analysis of the results

Based on the results shown in Fig. 3 and Fig. 4 and in Table 1., and Table 2., it is concluded that:

a) the mean estimated variance $\sigma_e^2$ curve for the LTWM algorithm is more closed to curve of true value, relative to other three compared algorithms. Mean absolute errors MAE of estimated variances is smaller value for all values $\alpha$. MSE of estimated variance of LTWM algorithm in relation to other algorithms is 6 (Alg. from [8]), 30.21 (Alg. from [6], [7]) and 3.86 (Alg. from [3]) times smaller.

b) in the range where the LTWM noise is not visible ($\alpha = 0 \div 10$), the mean estimated variance $\sigma_e^2$ curve for the LTWM algorithm is more closed to curve of true value, relative to other three compared algorithms. Mean absolute errors MAE of estimated variances is smaller value for all values $\alpha$. MSE of estimated variance of LTWM algorithm in relation to other algorithms is 5.71 (Alg. from [8]), 3.34 (Alg. from [6], [7]) and 2813.3 (Alg. from [3]) times smaller.

c) in the range where the LTWM noise is visible ($\alpha = 11 - 50$), the mean estimated variance $\sigma_e^2$ curve for the LTWM algorithm is more closed to curve of true value, relative to other three compared algorithms. Mean absolute errors MAE of estimated variances is smaller value for all values $\alpha$. MSE of estimated variance of LTWM algorithm in relation to other algorithms is 26 (Alg. from [8]), 30.2 (Alg. from [6], [7]) and 30.5 (Alg. from [3]) times smaller.

d) execution time of the LTWM algorithm in relation to other algorithms is 1.48 (Alg. from [8]), 21.77 (Alg. from [6], [7]), 34.82 (Alg. from [3]), times smaller.

Based on the conducted comparative analysis of the results, it is concluded that the proposed LTWM algorithm has the greatest precision in estimating the variance of the LTWM noise. In addition, the LTWM algorithm has the shortest execution time. The obtained results recommend the implementation of the LTWM algorithm in system for real-time regime.

6. Conclusions

The modelling of the noise, caused by the insertion of the watermark into the image by the LT algorithm, was performed in this paper. It was shown that the noise could be modelled by Gaussian distribution. The LTWM model was described, with which it was possible to estimate the parameters of LTWM noise ($\mu$, $\sigma^2$). The detailed analysis showed that mean absolute errors (MAE) of estimated variance of LTWM algorithm, proposed in this paper, in relation to other algorithms, is smaller in all
range of insertion factor. Further, the detailed analysis showed that mean square errors (MSE) in relation to the estimation accuracy in the application: a) Alg. from [8], b) Alg. from [6], [7] and c) Alg. from [3], was smaller than 6, 30.21 and 3.86 times, respectively. In relation to: a) Alg. from [8], b) Alg. from [6], [7] and c) Alg. from [3], it took less execution time $T_e$ for the LTWM algorithm, 1.48, 21.77 and 34.82 times, respectively. Based on the above mentioned results, it is concluded that the LTWM algorithm has high accuracy of estimation variance and short execution time. The mentioned performances give a recommendation for the implementation of the LTWM algorithm in systems for real-time regime.

References

[1] Dong L Zhou J and Dai T 2018 Color image noise covariance estimation with cross-channel image noise modeling IEEE International Conference on Multimedia and Expo (ICME).

[2] Ding F Shi Y Zhu G and Shi Y 2020 Real-time estimation for the parameters of Gaussian filtering via deep learning Journal of Real-Time Image Processing, Vol 17 pp. 17-27.

[3] Liu W and Lin W 2013 Additive white Gaussian noise level estimation in SVD domain for images IEEE Transactions on Image Processing Vol 22 Issue 3 pp. 872-883.

[4] Popp A Bock V Sedlmeir F C et al 2019 Investigation of noise sources down to the shot-noise limit in Yb-doped fiber amplifiers for TMI investigations Conference on Lasers and Electro-Optics Europe & European Quantum Electronics Conference (CLEO/Europe-EQEC).

[5] Russo F 2003 A method for estimation and filtering of Gaussian noise in images IEEE Trans. Instrum. Meas. Vol 52 Issue 4 pp. 1148 –1154.

[6] Liu X Tanaka M and Okutomi M 2012 Noise level estimation using weak textured patches of a single noisy image IEEE International Conference on Image Processing (ICIP).

[7] Liu X Tanaka M and Okutomi M 2013 Single-image noise level estimation for blind denoising noisy image IEEE Transactions on Image Processing Vol 22 Issue 12 pp. 5226-5237

[8] Garcia D 2010 Robust smoothing of gridded data in one and higher dimensions with missing values Computational Statistics & Data Analysis Vol 54 Issue 4 pp. 1167-1178.

[9] Yue H Liu J Yang J Nguyen T and Hou C 2017 Image noise estimation and removal considering the bayer pattern of noise variance IEEE Int Conf on Image Processing (ICIP).

[10] Ting F F Sim K S and Wong E K 2016 A rapid medical image noise variance estimation method International Conference on Robotics, Automation and Sciences (ICORAS).

[11] Kowalski J Mikolajczak G and Pekinski J 2018 Noise variance estimation in digital images using finite differences filter 41st Int Conf on Telecommunications and Signal Processing (TSP).

[12] Winarno A Setiadi D Arrasyid A Sari Rachmawanto C 2017 Image watermarking using low wavelet subband based on 8x8 sub-block DCT Int Seminar on Application for Technology of Information and Communication (iSemantic).

[13] Choudhary R and Parmar G 2016 A robust image watermarking technique using 2-level discrete wavelet transform (DWT) IEEE 2nd Int Conf on Communication, Control and Intelligent Systems pp. 120-124.

[14] Su Q Niu Y Liu X and Zhu Y 2012 Embedding color watermarks in color images based on Schur decomposition Optics Communications Vol 285 Issue 7 pp. 1792-1802.

[15] B. Prlincevic, Z. Milivojevic, P. Spalevic, and D. Brodic, "Performance of the SD algorithm for inserting watermark in image based on the schur decomposition" International Scientific Conference INFOTEH, Jahorina 2015, pp. 640-644, 2015.

[16] Milivojevic Z and Stevanovic Z 2013 Analysis on the robustness SVD-based watermarking algorithms Int. J. of Computer and Information Technology-JICIT Vol 2 Issue 04 pp. 688-693.

[17] Zong H Liu F-L and Luo X-Y 2012 Blind image steganalysis based on wavelet coefficient correlation Digital investigetion Vol 9 Issue 1 pp. 58-68.

[18] https://photogallery.sc.egov.usda.gov/photogallery/.