Image based Clustering Steganalysis on Multi-Projection Collection

V. Arunkumar
Research Scholar in Department of Computer Science,
Periyar University
Salem, Tamilnadu, India

Dr. K. Padmanabhan
Dean, Department of Computer Science & Applications,
Vivekanandha College of Arts & Sciences for Women,
Tiruchengode, Tamilnadu, India

***Abstract* - In this paper, we propose a novel algorithm called Multi-Projection Ensemble Discriminate Clustering (MPEDC) for Image Steganalysis. Propose to utilize the ideal projection of Straight Separate Examination (SSE) calculation to get more projection vectors by utilizing the smaller scale turn strategy. These vectors are like the ideal vector. MPEDC consolidates solo K-implies calculation to settle on an exhaustive choice grouping adaptively. The intensity of the proposed technique is shown on three steganographic strategies with three component extraction techniques. Test results show that the precision can be improved utilizing iterative segregate characterization.**

Key words: Image Steganalysis, Multi-Projection, Linea Distinguish Analysis, K-Means

I. INTRODUCTION

The conflict between Steganography & Steganalysis has come to be fundamental impenetrability of Information Security. The most popular steganalysis mostly consists of function extraction and classifier learning. For JPEG images, early aspects are at once used in the DCT area to instruct the classifier. The CC-JRM (JPEG wealthy model with Cartesian-calibration) [1] characteristic uses the concept of feature fusion to fuse the 40 sub-models of inter block and intra-block statistical houses of DCT model and the subset of eleven sub-models of DCT essential co-occurrence matrices. Later, the researchers proposed Discrete Cosine Transform Residual (DCTR) [2] and Gabor Filter Residual (GFR) [3], which are higher feature extraction methods. In this paper, we will use these traditional aspects to instruct the tremendous classifier.

As for characteristic classification, there exists a large range of various computing device to gaining knowledge tools employed in steganalysis. The Linear Discriminate Analysis (LDA) ensemble classifier [4] can maintain a quick running velocity underneath excessive dimensional fact & exact accuracy. It carries multiple LDA sub-classifiers & each sub-classifier randomly extracts a part of the function to assemble the feature subspace. In our model, we combine LDA & K-Means clustering [5], [6] to greater precisely address the complicated issues arising in steganalysis, and make use of ensemble learning to create the uniform detection model. Self-learning ensemble discriminate clustering [7] effectively utilized the ensemble gaining knowledge of idea to create an ensemble classifier consisting of LDA and k-means classifiers trained on a set of Stego and cover images to remedy the problems of steganalysis in excessive dimensional characteristic space.

II. BACKGROUND

Both LDA & K-means are elegant first-class cataloging strategies in Machine Learning. The built-in classifier which combines LDA and K-means can resolve stego free image steganalysis problems.

A. LDA and K-Means

The Linear Discriminate Analysis (LDA) is one of the most generally utilized segregation basis in the component grouping, which characterizes a projection vector that makes the inside class disperse S_w littler and the between-class dissipate S_b bigger. The LDA technique can well decrease the dimensionality of picture highlights, and it has a solid intensity of separation which is broadly used to choose the element subspace. K-implies calculation, as a hard grouping calculation, is a run of the mill illustrative of the model based target work bunching strategy utilizing the iterative change rules.

B. Self-Learning Ensemble Discriminate Clustering

Self-learning Ensemble Discriminate Clustering is denoted as SEDC in [7], where the typical of every sample point is projected onto the vector obtained by LDA and used because the initial cluster center of the K-means algorithm. The simplest projection direction w is given which is defined by max J(w)as follows:

$$\max_{\mathbf{w}} \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}, \qquad (1)$$

To obtain max J(**w**), we minimizes S_w, and maximizes S_b. **w** can be calculated by

$$\mathbf{w} = \mathbf{S}_w^{-1}(\mathbf{u}_1 - \mathbf{u}_2) \qquad (2)$$

Where u1 and u2 are the means of the cover and stego features.

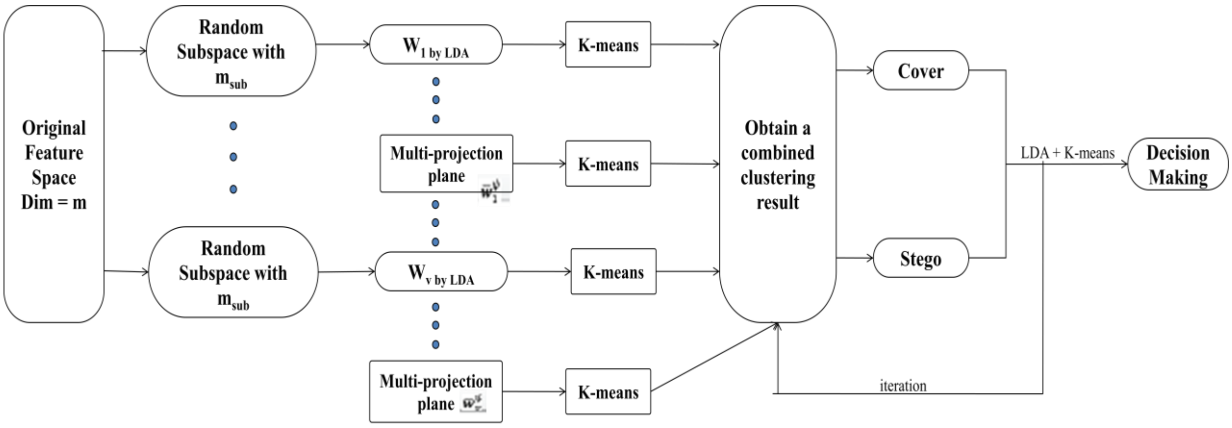

Fig. 1 Block diagram of the proposed MPEDC. The random subspace of each example is constructed by sampling $m_{sub} << m$ feature randomly from the entire feature space.

III. MULTI-PROJECTION ENSEMBLE DISCRIMINATE CLUSTERING

The MPEDC (Multi-Projection Ensemble Discriminate Clustering) additionally incorporates the LDA and K-implies calculations. For the decent variety of separate, we attempt to display the arbitrary dispersion in the classifier and search for a multi-projection course. The separated highlights are utilized to prepare various sub-classifiers. Figure 1 gives the general square graph of the proposed MPEDC.

A. Problem Formulation

We note a classification problem with n_{tr} training samples and feature with m-dimension. For $m_v \subset \{1,...,m\}$, $|m_v| = m_{sub}$ is the number of random subspace. In the vth sampled subset, LDA is trained $\{\mathbf{x}_1^{\mathbf{m}_v}, \ldots, \mathbf{x}_i^{\mathbf{m}_v}\}_{i=1}^{n_{tr}}$, and tested on $\{\mathbf{y}_1^{\mathbf{m}_v}, \ldots, \mathbf{y}_j^{\mathbf{m}_v}\}_{j=1}^{n_{te}}$, where $\mathbf{x}_i^{\mathbf{m}_v}$ is m_{sub} msub-dimension feature from the original cover feature x_i and $\mathbf{y}_i^{\mathbf{m}_v}$ is m_{sub} dimension feature from the original feature y_j to be detected, which are batch unlabeled feature data. n_{tr} and n_{te} are the numbers of training and testing samples. We use u1 and u as means of the cluster in the cover and the unlabeled testing feature, respectively. In the vth subspace, the projection vector w_v is obtained by LDA on the training set.

The centroid of each class u_j means u_j = $\sum_{\mathbf{y}_j^{\mathbf{m}_v} \in \emptyset_j} \mathbf{y}_j^{\mathbf{m}_v}$ /nj , (j = 1, 2), where n1 and n2 are the number of cover and stego images to be detected (unknown) [7] and ∅j is the jth cluster. For MPEDC, we assume that the input cover images have labels and the cover images to be detected have the same statistical property, e.g. u1 ≈ u1. Therefore, u2 can be expressed as

$$\bar{\mathbf{u}}_2 = \bar{\mathbf{u}} - \bar{\mathbf{u}}_1 \approx \bar{\mathbf{u}} - \mathbf{u}_1 \quad (3)$$

Now, in the vth view, total scatter matrix S_t and between class scatter matrix S_b may therefore be expressed as:

$$\bar{\mathbf{S}}_t = \sum_{i=1}^{n} (\bar{\mathbf{y}}_i^{\mathbf{m}_v} - \bar{\mathbf{u}})(\bar{\mathbf{y}}_i^{\mathbf{m}_v} - \bar{\mathbf{u}})^T \quad (4)$$

$$\bar{\mathbf{S}}_b = \sum_{j=1}^{2} n_j (\bar{\mathbf{u}}_j - \bar{\mathbf{u}})(\bar{\mathbf{u}}_j - \bar{\mathbf{u}})^T \quad (5)$$

Where the total scatter matrix S_t means $S_t = S_b + S_w$. Fig. 1, we need to get the vth projection vector w_v, here,

$$\bar{\mathbf{w}}_v = \bar{\mathbf{S}}_w^{-1}(\bar{\mathbf{u}}_1 - \bar{\mathbf{u}}_2) = (\bar{\mathbf{S}}_t - \bar{\mathbf{S}}_b)^{-1}(\bar{\mathbf{u}}_1 - \bar{\mathbf{u}}_2) \quad (6)$$

B. Multi-Projection Accessing & K-Means Clustering

MPEDC calculation small scale pivot the projection vector w_v which are like w_1, and afterward venture the examples onto various vectors approximating the best projection vector for coordinated order to get increasingly precise grouping results. w is as per the following

$$\bar{\mathbf{w}}_v^{\psi} = \begin{cases} \bar{\mathbf{w}}_v^{\psi} + \mathbf{a}^{\psi} .* \bar{\mathbf{w}}_v^{\psi}, & \psi > 0 \\ \bar{\mathbf{w}}_v, & \psi = 0 \end{cases} \quad (7)$$

where w_v is the ψth projection vector obtained randomly from w_v. The operation .∗ means the element-by-element multiplication. ψ is a positive integer, which is a parameter related to embedding rate r expressed as

$$\psi = \begin{cases} 5 - round(10r), & r \le 0.5 \\ 0, & r > 0.5 \end{cases} \quad (8)$$

where ψ and r are negatively correlated and 10r should be an integer. If 10r is not an integer, we will round off this value. According to the LDA algorithm, the **a** stands for a randomly vector containing either positive or negative elements with values close to zero. Therefore, a is defined as

$$\mathbf{a} = \frac{2\mathbf{b} - 1}{10^{10r-1}} \quad (9)$$

where b is used to generate a random vector of m_{sub} dimensions with element values between 0 and 1.

When calculating a^{ψ}, we can get the corresponding $\bar{\mathbf{w}}_v^{\psi}$. About the choice of parameter ψ, we will explain more specification in the experimental part of the article.

After obtaining multiple projections, MPEDC can project $\bar{\mathbf{u}}_1$ and $\bar{\mathbf{u}}_2$ of each sub-classifier onto the corresponding projection vector respectively as the first cluster center of K-means clustering, i.e., $\bar{\mathbf{u}}_1 \bar{\mathbf{w}}_v^{\psi} \approx \mathbf{u}_1 \bar{\mathbf{w}}_v^{\psi}$. Every instance nearest to the clustering centroid will be distributed to the corresponding class.

In each sub-classifier, there will be two categories of cover and stego. MPEDC will re-cluster them with LDA and K-means algorithms, which means these two categories using LDA are projected onto a single vector for the supervised classification. The pseudo code of the iteration process is presented in Algorithm 1, where the parameter T is the number of iterations. The above-mentioned algorithm is shown in Algorithm 2. The parameter L stands for the number of the sub-classifier. In particular, P_E and τ represents the detection error and the number of experiments, respectively.

IV. EXPERIMENTAL VERIFICATION

In our experiments, a total of 10,000 JPEG grayscale images from the BOSS BASE 1.01 [8] with the same size 512 × 512 and quality factors QF = 55 and QF = 85 are used as the unique covers. We performed nsF5 (no-shrinkage F5) [9] and J-UNIWARD[8] steganographic methods on the original images to produce 10,000 stego images using CC-JRM[1], DCTR [2] and GFR [3]. All the results are from the average of $\tau = 10$ times.

ALGORITHM A

Iteration Process

1: for T ← 1 to t do
2: Get tagged cover and stego images according to the previous classification results;
3: Compute the best projection vector with LDA algorithm by Eq. (2);
4: Run K-means: obtain the cluster label vector;
5: end for

ALGORITHM B

The proposed Clustering Algorithm
Ensure: Cluster label vector and P_E
1: for meantime ← 1to τ do
2: Form a random subspace $m_v \subset \{1,\dots,m\}$, $|m_v| = m_{sub} << m$
3: for l ← 1 to K do
4: Compute u1, u, u2, St , Sb by Eqs. (3) – (5)
5: Compute the projections w_v by Eq. (6)
6: Compute rotated multi-projection w_ψ v by Eqs. (7) – (9)
7: Compute the projection vector for all samples & their means, such as $\mathbf{u}\bar{\mathbf{w}}_v^{\psi}$, $\mathbf{Y}\bar{\mathbf{w}}_v^{\psi}$
8: Run K-means: obtain the cover, stego clusters and their label vectors
9: Run iteration algorithm in Algorithm A
10: end for
11: end for

A. Detection Error Comparisons

In Tables 1–2, we can obviously see the error rates of detecting the features of different steganalysis methods in J-UNIWARD and nsF5 with different embedding rates. For example, the error detection rates of MPEDC for different embedding rates of DCTR features in J-UNIWARD are almost lower than SEDC as QF = 75. However, the CCJRM features with different embedding rates show different characteristics, and the detection rate of SEDC algorithm is lower than that of MPEDC at the embedding rates of 0.2, 0.3
and 0.4, which are respectively 47.9%, 41.4% and 33.7%. With the higher embedding rate, the detection is easier, especially against nsF5. Also, both SEDC and MPEDC methods have the poor performance on J-UNIWARD with lower embedding rates.

From Tables 1–2, we can clearly see that there are a few results that MPEDC is lower than SEDC. When calculating the rotating multi-projection vector, b is a random vector, so the vector obtained by the rotation has a certain randomness, which may lead to a very small number of cases that have a negative impact on the classification result. Even if our experiment takes the average of 10 experiments ($\tau = 10$), the negative effects cannot be completely excluded. Moreover, most of the classifiers do not have a good classification effect on the features of low embedding rate, and the MPEDC algorithm will amplify the negative effects on the features of low embedding rate. As shown in Table 1, when the embedding rate is 0.1 for the GFR (QF = 75), the error detection rate of MPEDC is higher than SEDC by 2.6%.

Table 1: The detection errors for different steganalysis schemes using SEDC and MPEDC in J-UNIWARD of different payloads with QF = 75 and QF = 95

QF	Feature	Method	Payload (bpnzac)			
			0.1	0.2	0.3	0.4
55	CC-JRM	SEDC	54.1%	48.9%	42.4%	34.7%
		MPEDC	53.4%	20.4%	43.5%	35.4%
	DCTR	SEDC	53.3%	45.0%	33.1%	25.1%
		MPEDC	52.7%	44.4%	33.0%	22.0%
	GFR	SEDC	47.7%	39.1%	25.4%	19.1%
		MPEDC	50.3%	37.8%	25.2%	15.0%
85	CC-JRM	SEDC	55.1%	55.0%	55.8%	48.2%
		MPEDC	54.0%	53.7%	52.5%	48.2%
	DCTR	SEDC	54.1%	54.0%	49.9%	44.4%
		MPEDC	55.3%	53.4%	49.5%	43.9%
	GFR	SEDC	52.7%	51.7%	46.8%	39.9%
		MPEDC	54.1%	51.4%	44.2%	36.3%

Table 2: The detection errors for different steganalysis schemes using SEDC and MPEDC in nsF5 of different payloads with QF = 75 and QF = 95

1QF	Feature	Method	Payload (bpnzac)			
			0.05	0.1	0.15	0.2
55	CC-JRM	SEDC	46.7%	26.5%	18.6%	11.5%
		MPEDC	44.9%	28.0%	16.5%	8.7%
	DCTR	SEDC	45.0%	32.6%	20.1%	14.1%
		MPEDC	48.7%	33.9%	17.6%	9.0%
	GFR	SEDC	48.4%	37.6%	27.1%	18.7%

		MPEDC	49.1%	36.7%	24.4%	15.3%
85	CC-JRM	SEDC	40.5%	22.3%	13.3%	6.7%
		MPEDC	41.4%	19.7%	7.8%	3.0%
	DCTR	SEDC	46.7%	30.7%	17.1%	8.4%
		MPEDC	49.1%	28.4%	13.0%	4.6%
	GFR	SEDC	50.8%	37.2%	28.4%	19.9%
		MPEDC	48.4%	38.7%	26.1%	16.0%

Table 3: For the different features of the four embedding rates, as shown in the 12 sets of experiments in Tables 1–2, improving AVE of the detection rate of MPEDC compared to SEDC

QF	J-UNIWARD			nsF5		
QF	CC-JRM	DCTR	GFR	CC-JRM	DCTR	GFR
65	−0.55%	1.10%	0.65%	1.20%	0.55%	1.48%
85	0.658%	0.65%	1.18%	2.63%	1.85%	1.68%

In Table 3, we list the improved average error detection rate (AVE) of MPEDC relative to SEDC under the four embedding rates of the same feature. It can be seen that for the 12 sets of experiments in Tables 1–2, AVE corresponding to the J-UNIWARD QF of 75 is slightly worse, and the other 11 sets of experiments are greatly improved, which also proves the effectiveness of our approach.

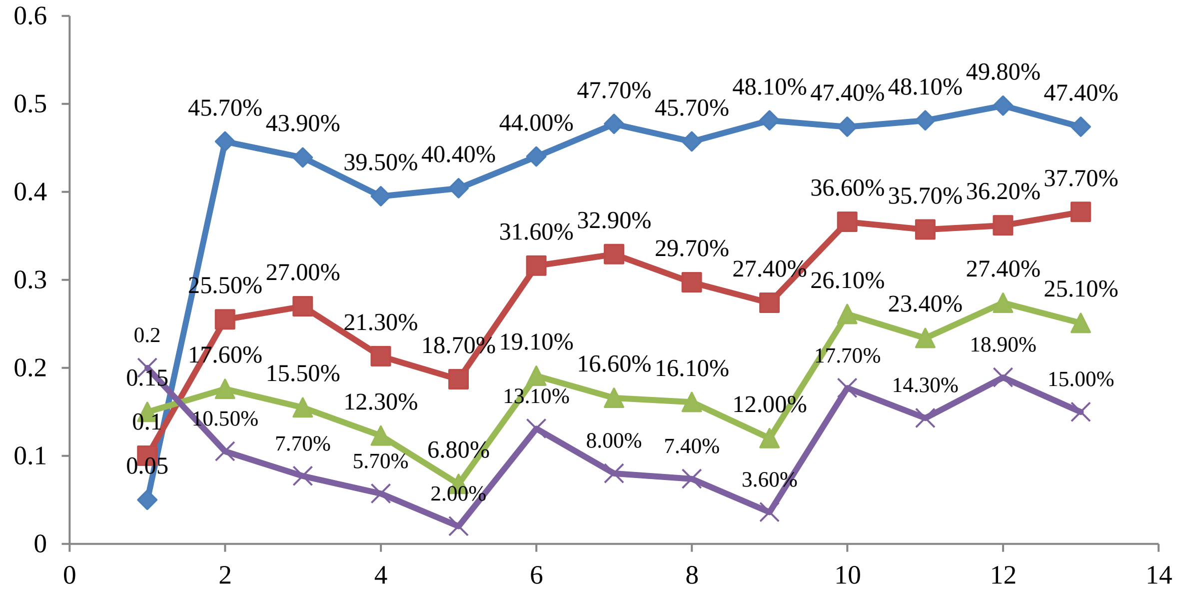

Fig. 2 For QF = 75, PE over ten iterations of DCTR against J-UNIWARD (payload = 0.5), GFR against J-UNIWARD (payload = 0.4), and CC-JRM against nsF5 (payload = 0.1) with different dims of sub-classifiers, where iteration L, τ and msub are 95, 10 and 1100, respectively

B. Iterative Weight Definition

In Fig. 2, we can clearly see that when the three of features DCTR, GFR and CC-JRMare in the first iteration, the detection error rate is reduced by a large margin, while in more than second iterations, although the error rate is reduced, the reduction rate is less. Considering the time complexity and efficiency of our classification, we think that the performance of the classifier is higher when the number of iterations T is 1.

C. The Selection of Parameter ψ

The size of ψ has an important relation with the embedding rate as Eq. (8). For example, when the embedding rate is 0.2, the projection rotates slightly three times; when the embedding rate is greater than or equal to 0.5, the projection does not rotate. In Eq. (8), 10r should be an integer. When 10r is not an integer, we round off the r value. For example, when the embedding ratio is 0.015, the value of 10 × 0.015 is 0.15, and then we take the approximation of r as 0.2 and the number of ψ as 3.

V. CONCLUSION

In this paper, we describe to the comfy association between LDA and K-means clustering. At that point, turn a projection acquired by the LDA calculation in an arbitrary subspace and yield roughly numerous projections to consolidate LDA and K-implies grouping into MPEDC. Exploratory outcomes show that the proposed strategy can viably identify J-UNIWARD and nsF5 as the best in class steganographic calculation. Particularly for steganographic highlights with a high implanted rate, the identification mistake rate is lower.

VI. REFERENCES

[1] J. Kodovsk´ y and J. Fridrich, "Steganalysis of JPEG images using rich models" Proceedings of SPIE, Electronic Imaging, Media Water-marking, Security, and Forensics XIV, San Francisco, CA, Jan. 23–25, vol.8303, p.83030A, 2012.
[2] V. Holub and J. Fridrich, "Low complexity features for JPEG steganalysis using undecimated DCT," IEEE Trans. Inf. Forensics Security, vol.10, pp.219–228, 2015.
[3] X. Song, F. Liu, C. Yang, X. Luo, and Y. Zhang, "Steganalysis of adaptive JPEG steganography using 2D Gabor filters,"

Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security, pp.15–23, 2015.

[4] J. Kodovsk´ y, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," IEEE Trans. Inf. Forensics Security, vol.7, no.2, pp.432–444, 2012.

[5] C. Ding and T. Li, "Adaptive dimension reduction using discriminant analysis and K-means clustering," Proceedings of the 24th International Conference on Machine Learning (ICML), pp.521–528, 2007.

[6] A. Wu, G. Feng, X. Zhang, and Y. Ren. "Unbalanced JPEG image steganalysis via multiview data match," Journal of Visual Communication & Image Representation, vol.34, pp.103–107, 2016.

[7] B. Cao, G. Feng, Z. Yin, and L. Fan, "Unsupervised image steganalysis method using self-Learning ensemble discriminant clustering," IEICE Trans. Inf. & Syst., vol.E100-D, no.5, pp.1144–1147, 2017.

[8] V. Holub, J. Fridrich, and T. Denemark, "Universal distortion function for steganography in an arbitrary domain," EURASIP Journal on Information Security, vol.2014, no.1, pp.1–13, 2014.

[9] J. Fridrich, T. Pevn. y and J. Kodovsk. y, "Statistically undetectable JPEG steganography: Dead ends, challenges, and opportunities," Proceedings of 9th ACM workshop on Multimedia & security, Dallas, TX, Sept. 20–21, pp.3–14, 2007.