Detection and Classification of Blur Images using Multi-Class Support Vector Machine

S Lakshmi Bhavani

dept. of Electronics and Communication Engineering
JNTUK, Vizianagaram, India

M Hema

Assistant Professor of dept. Electronics and
Communication Engineering
JNTUK, Vizianagaram, India

Abstract— **In recent technology, it has been critical for blind image restoration. It is focused on the blur classification of digital images using a Multi-class Support Vector Machine (MSVM) structure. This work aims to classify the blur images using MSVM. MSVM classifier is designed to identify three types of images like Sharp, Defocused, and Motion blurred images. Several experiments are conducted for a sample data called Beihang Univ. Blur Image Database (BHBID). The Mean, Variance, and maximum edge detected feature matrix are taken for each image applied on Sobel, Laplacian, and Roberts cross edge detections. Based on sampling, features are selected to train each member of the MSVM classifier. Using different kernels of SVM's like Linear, Polynomial, Radial Basis Function (RBF), Gaussian, it can optimize the parameters, and the performance metrics like accuracy will be compared. Finally, our proposed system achieved 95.7 % accuracy in finding the defined scenarios**

Keywords— Blur Image Classification, SVM-RM, MSVM, Edge Detection by Sobel and Laplacian operators, Feature selection

I. INTRODUCTION

Blur Image, is a form of bandwidth reduction of a normalized image owing to an imperfect formation process, and it is the primary source that leads to image degradation. This causes blur-incurring point spread function (PSF) in a simulated environment, out-of-focus of the imaging system, and target motion during the signal capturing process [1]. Blur images are categorized into three main branches like blur detection and classification, along with image restoration. In general, image quality can be classified into subjective or objective techniques. The subjective technique of image quality is costly and time-consuming, and the outcome would also depend on viewing conditions [2]. The non-blind methods [3] [4] require prior knowledge of the blur kernel parameters, whereas the blurring operators are assumed to be unknown in advance in the blind methods [5]. Deblurring a blurred image without PSF using blind methods is more challenging. For instance, single-channel blind deconvolution within the Bayesian framework is proposed in [6] and the multiple scattering model-based remote sensing image restoration methods in [7]. In addition to the image deblur, blur detection and blur classification are critical to image deblurring issues that are increasingly attractive in image processing. From [8] [9] [10], the information of blur parameters is necessary for blur image recovery, which is obtained from blur detection and blur classification.

Another blur detection technique is an extrema analysis of the image in the spatial domain [11], a Laplacian, Sobel, Roberts-Cross-Edge-Detection. After the Blur detection, we need to classify the blur images. Here we are using the Multiclass Support Vector Machine (MSVM) to classify the

blurred images. The statistical learning theory in the year 1960 was developed for learning algorithms of nonlinear functions by the seminal work with Vapnik and Chervonenkis [12]. The multiclass classification with SVM is ongoing research. In this, we primarily focus on a novel SVM designed to classify multiclass and high dimensional datasets [13]. We implemented SVM and averaged the results to compare with different kernels with the classic One-Vs-One method regarding prediction accuracy. So we are using MSVM to classify three types of images: Sharp, Defocused and Motion blurred images.

Section II introduces the proposed algorithm and the system formulation based on image detection using Laplacian, Sobel, Roberts cross edge detection. And classification using MSVM. In sections III-A, III-B, III-C, we briefly explain the detection techniques of blur type images using Laplacian, Sobel, Roberts cross edge detection, respectively. In section IIID, we briefly describe the Multi-class Support Vector Machine classification. In section IV, experimental results were discussed, and finally, the paper is concluded with achievements and further possible extensions for the future works in section V.

II. SYSTEM DESIGN

In Fig. 1, the blur image classification is illustrated. Here, we have taken clear images and also blurred images like Motion, Defocused are taken as input images. In the spatial domain, the blur detection technique is an extreme analysis of the image. Using Laplacian, Sobel, Roberts-Cross-Edge-Detection techniques, we detect edges of the Blurred images, and the necessary features are extracted from the edge detection techniques. These extracted features are given as an input to the Support Vector Machine Algorithm for the classification of Blur Images. By using different kernels of SVM like Linear, Polynomial, Radial Basis Function (RBF), it can classify and optimize the parameters and the performance metrics like accuracy, precision, and recall will be compared.
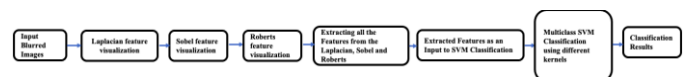

Fig. 1. Detection and Classification of Blur Images Block Diagram

III. BLUR FEATURES EXTRACTION

A. *Laplacian Operator*

Fig. 2 represents the Laplacian edge visualization of the blur image dataset. To find edges in a digital image, we use one edge detection operator known as the Laplacian operator. Out of all the edge detection operators like Prewitt, Sobel etc. The

Laplacian operator is a second-order derivative mask. The formulae for Laplacian function f(x, y) is

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

(1)

The second derivatives along x and y-direction can be approximated as

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial G_x}{\partial x}$$

$$\circ \quad = f[i, j+2] - 2f[i, j+1] + f[i, j]$$

$$\circ$$

(2)

The approximation is centred about the pixel [i, j+1]. So replacing j with j-1, we obtain.

$$\frac{\partial^2 f}{\partial x^2} = f[i, j+2] - 2f[i, j] + f[i, j-1]$$

(3)

Similarly

$$\frac{\partial^2 f}{\partial y^2} = f[i+1, j] - 2f[i, j] + f[i-1, j]$$

(4)

By combining the above two equations into a single operator, the following mask can be approximate the Laplacian : ∇^2

Mean for the Laplacian operator $\mu_L = \mu(\nabla^2)$

The variance of Laplacian operator $\sigma_L = \sigma(\nabla^2)$

Maximum of Laplacian operator $max_L = max(\nabla^2)$

TABLE I   MASK OF LAPLACIAN OPERATOR

0	1	0
1	-4	1
0	1	0

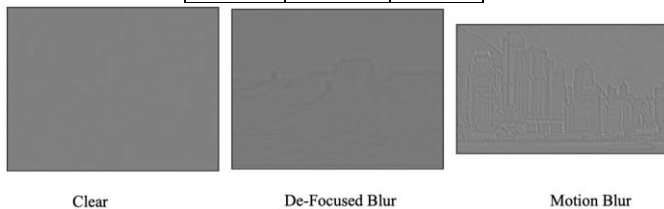Clear   De-Focused Blur   Motion Blur

Fig. 2. Laplacian edge feature visualization

B. Sobel Operator

Fig. 3 represents the Sobel edge visualization of the blur image dataset. Irwin Sobel proposed a technique which is a Sobel edge detection technique [14] [15] [16] in 1970. One of the pixel-based edge detection algorithms is Sobel Operator. Edges were detected by calculating partial derivatives in 3 x 3 neighbourhoods. We are using the Sobel operator to detect edges because it is not sensitive to noise, and it is relatively a small mask in images. TABLE II shows the convolution kernel, which is simply rotated by 90^0 And these kernels are designed to detect edges vertically and horizontally relative to the pixel grid of images, concerning one kernel for each of the two

perpendicular orientations. To obtain the absolute magnitude of the gradient at each point, the kernels can be applied separately to the input image to produce a separate measurement of the gradient component in each orientation. The Partial derivatives in the x and y direction are given as follows:

$$S_x = \{f(x+1, y-1) + 2f(x+1, y) + f(x+1, y+1)\} - \{f(x-1, y-1) + 2f(x-1, y) + f(x-1, y+1)\}$$

(5)

$$S_y = \{f(x-1, y+1) + 2f(x, y) + f(x+1, y+1)\} - \{f(x-1, y-1) + 2f(x, y) + f(x+1, y-1)\}$$

(6)

The gradient of each pixel is calculated using:

$$g(x, y) = \sqrt{S_x^2 + S_y^2}$$

(7)

TABLE II   MASK OF SOBEL OPERATOR

-1	0	+1		+1	+2	+1
-2	0	+2		0	0	0
-1	0	+1		-1	-2	-1

Mean for the Sobel operator $\mu_s = \mu(g(x, y))$

The variance of Sobel operator $\sigma_s = \sigma(g(x, y))$

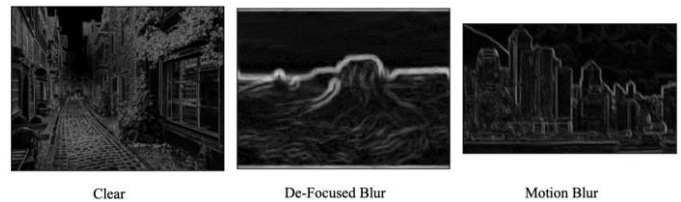Maximum of Sobel operator $max_s = max(g(x, y))$

Clear   De-Focused Blur   Motion Blur

Fig. 3. Sobel edge feature visualization

C. Robert Cross Edge Operator

The Robert Cross operator performs a quick 2-D spatial gradient detection on an image. It consists of a pair of 2 x 2 convolution kernels as in TABLE III. These kernels are implemented to respond maximum to edges running at 45∘ to the pixel grid one kernel for each of the two perpendicular orientations. The kernels can be applied to the input image to get a separate measurement of the gradient component in each orientation then these are combined to find the absolute magnitude of the gradient at each point, and the direction of the gradient is represented by:

$$localization = \frac{1}{\sqrt{E[x_0^2]}} = \frac{\left| \int_{-w}^{w} G'(-x) f'(x) dx \right|}{\sqrt[n_0]{\int_{-w}^{w} f'^2(x) dx}}$$

(8)

TABLE III   MASK OF ROBERT CROSS EDGE OPERATOR

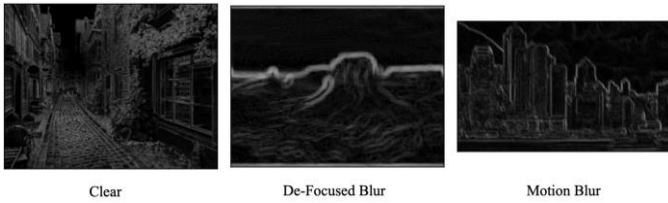| +1	0		0	+1
0	-1		-1	0

Clear   De-Focused Blur   Motion Blur

Fig. 4. Robert cross edge feature visualization

Mean for the Sobel operator $\mu_R = \mu(\lfloor G \rfloor)$

The variance of Sobel operator $\sigma_R = \sigma(\lfloor G \rfloor)$

Maximum of Sobel operator $max_R = max(\lfloor G \rfloor)$

After considering the mean, variance and maximum values of all the above edge detection operators, these nine values take a feature vector for MSVM to classify clear, motion defocused blurred images.

$$Feature's\ Vector = [(Laplacian\ Mean, Var, MaxVal),$$
$$(Sobel\ Mean, Var, MaxVal), (Robert\ Mean, Var,$$
$$MaxVal)]$$

(9)

D. Multi-class Support Vector Machine

A discriminative classifier takes training data (supervised learning), and the algorithm gives an optimal hyperplane. The SVM consisting a maximum margin, and the width of the margin is given as

$$\frac{2}{\| w \|}$$

(10)

Our goal is to maximize the margin. Now the quadratic programming problem is

$$\min_{w \in R^m} \frac{1}{2} \| \mathbf{w} \|^2$$

(11)

Here w is an average vector called weight which controls the direction of the hyper-plane. Here, the data we took is linearly separable and can be performed on a higher dimensional vector space. The corresponding optimization in feature space is

$$max \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \Phi(\mathbf{x}_i) \Phi(\mathbf{x}_j)$$

(12)

Here x_i And x_j are support vectors (data points) on the hyperplane. y_i, set of class labels such that α is an arbitrary constant and $\emptyset$ is mapping from input space to feature space. The final classification is taken as,

$$bias + \sum_{s \varepsilon SV} w_s K(x^{test}, x^s) > 0$$

(13)

Here SV is a set of support Vectors. Here we need to select the support vectors that maximize the margin and compute the weight on each support vector. Here we are choosing a different

kernel function of SVM, and we are comparing the accuracy performance of three classes

The linear kernel is the most straightforward kernel function. An SVM built with the linear kernel is generally equivalent to its non-kernel counterpart. c is an arbitrary constant

$$k(x_i, x_j) = x_i^T x_j + c$$

(14)

The polynomial kernel is a non-stationary kernel function. It is suitable for problems in which the training is normalized [17] [18] [19]. α is the slope. d is the degree of the polynomial, and c is an arbitrary constant

$$k(x_i, x_j) = (\alpha x_i^T x_j + c)^d$$

(15)

RBF Kernel is a versatile and efficient kernel function. It can be computationally expensive for a high dimensional input space. γ is an adjustable parameter.

$$k(x_i, x_j) = \exp(-\gamma \| x_i - x_j \|^2)$$

(16)

Since Support Vector Machine (SVM) can deal only with binary classification problems, our objective of classifying three different blur image datasets cannot be handled by binary SVM. There are modified SVMs also for multi-class problems such as One Vs One and One Vs Rest. These two methods are used to classify multi-class problems into a fixed number of binary classes. Unlike One Vs One and One Vs Rest, the Error-Correcting Output Codes (ECOC) technique is used to divide each class that is to be encoded as an arbitrary number of the binary classification problem. Here we are using the One Vs One method for the ECOC technique. The one-vs-one method constructs k(k−1)/2, given that k is the number of classes [17, 19, 20].
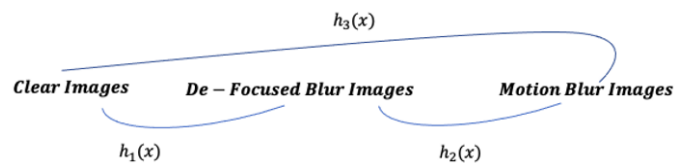

Fig. 5. One Vs One Hypothesis

So in Fig. 5, there are three hypotheses. Apply all of these hypotheses one by one to the input class X and check which class will get majority voting will be that class of X.

E. Error-Correcting Output Codes (ECOC) Method

In ECOC, we will assign binary codes corresponding to each class and make sure that we cannot assign the same binary codes for each class as shown in TABLE IV. Depending on the length of the code, we are going to define the

Published by :

http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 10 Issue 11, November-2021

TABLE IV   ERROR-CORRECTING OUTPUT CODES (ECOC) METHOD

	Learner1	Learner2	Learner3
Class 1(Clear Images)	0	1	0
Class 2 (de-focused Images)	0	0	1
Class3 (Motion Blur Images)	1	0	0
New Instance	1	0	0

Binary learners. The binary Learners uses different Algorithms like Native-Bays as the algorithm is implemented in MATLAB; it uses the default Learner Template (LT). Checking the greatest similarity or minimum distance by calculating differences of binary codes of learners of each class with new instance class, new instance class is going to be classified as the class which is going to have the greatest similarity or minimum distance. If we increase the length of the Learners, then it is going to increase the classifier performance.

IV. EXPERIMENTAL RESULTS

The proposed model is conducted over a sample dataset which Beihang Univ. Blur Image Database (BHBID). The training dataset of simulation blurred image dataset has 100 defocus blur, 100 motion blur, and 100 clear images for 300 samples. The testing dataset of Simulation Blurred images contains 100 defocus blur, 100 motion blur, and 100 clear images for a total of 300 samples. Here the training and testing data sets are different images. Here we take an equalized number of images for training and testing for all classes to have unbiased training. Using MATLAB, we process the algorithm to classify the testing images as clear, defocused or blurred images. Using the confusion matrix, the performance of a classification model or classifier on a set of test data for which the actual values are known. By knowing True Positive(TP) and False Positive(FP) in the confusion Matrix, the accuracy of classification is calculated, and the formulae for calculating accuracy of classification is:

$$Accuracy = TP + TN/TP + FP + FN + TN$$

(17)

Here TP = True Positive of Confusion Matrix

TN = True Negative of Confusion Matrix

FP = False Positive of Confusion Matrix

FN = False Negative of Confusion Matrix

Using different kernels of Support Vector Machine (SVM) as described in the above session, which can optimize the parameters and the performance metrics like accuracy, is compared.

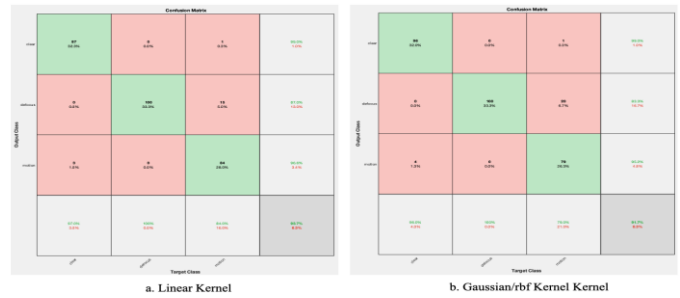a. Linear Kernel   b. Gaussian/rbf Kernel Kernel

Fig. 6. Confusion Matrix of MSVM

The above confusion matrix consists of three classes clear, defocused, motion. These three classes consist of three hundred images by Multi-class Support Vector Machine algorithm is predicting and classifying the testing set images. Here we are using the different Kernels to compare and calculate the performance accuracy.
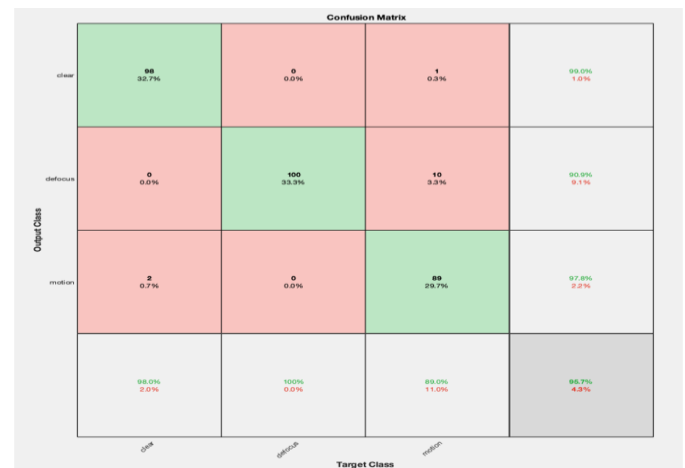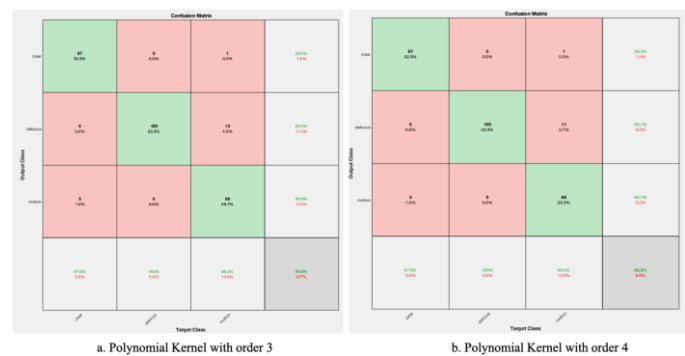

a. Polynomial Kernel with order 3   b. Polynomial Kernel with order 4

c. Polynomial Kernel with order 6

Fig. 7. Confusion Matrix of MSVM

TABLE V   ACCURACY PERFORMANCE OF MSVM CLASSIFICATION

SVM Kernel Functions	Accuracy Percentage
Linear Kernel	91.7
Gaussian/RBF Kernel	93.7
Polynomial Kernel with Order 3	94.2
Polynomial Kernel with Order 4	95
Polynomial Kernel with Order 6	95.7

V. CONCLUSION

In this paper, we investigated the blur images classification using a Multiclass Support Vector machine. Also, we addressed and compared the classification accuracy by using different kernels of SVM. However, with the aim of a machine learning classifier, namely Multiclass Support Vector Machine, high accuracy was obtained to determine the blurred image classes. Therefore, by defining efficient features, the accuracy can be enhanced.

REFERENCES

[1] J.H. Elder, S.W. Zucker, Local scale control for edge detection and blur estimation, IEEE Trans. Pattern Anal. 20 (7) (1998) 699–716.

[2] Cohen, E, Yitzhaky, Y., "No-Reference assessment of blur and noise impacts on image quality," published in Springer London, Signal, Image and Video Processing, 2009.

[3] D. Krishnan, R. Fergus, Fast image deconvolution using hyper-laplacian priors, supplementary material, in: Neural Information Pro-cessing Systems Conference. 2009.

[4] H.Hong,Y.Shi,Fastdeconvolutionformotionbluralongtheblurringpaths,Canad. J. Electr. Comput. Eng. 40 (4) (2017) 266–274

[5] M.S.Almeida,L.B.Almeida,Blindandsemi blind-de-blurring of natural images,IEEE Trans. Image Process. 19 (1) (2010) 36–52.

[6] A.C. Likas, N.P. Galatsanos, A variational approach for bayesian blind image deconvolution, IEEE Trans Signal Process. 52 (8) (2004) 2222–2233

[7] S. Tao, H. Feng, Z. Xu, Image degradation and recovery based on multiple scattering in remote sensing and bad weather condition, Opt. Express 20 (15) (2012) 16584– 16595.

[8] Y.J. Li, X.G. Di, Image mixed blur classification and parameter identification based on cepstrum peak detection, in Control Conference, CCC, 2016 35th Chinese, IEEE, 2016, pp. 4809–4814.

[9] R. Wang, W. Li, R. Qin, J. Wu, Blur classification based on deep learning, in Imaging Systems and Techniques, IST, 2017 IEEE International Conference on, IEEE, 2017, pp. 1–6.

[10] D. Yang, S. Qin, Restoration of the degraded image with partial, blurred regions based on blur detection and classification, in Mechatronics and Automation, ICMA, 2015 IEEE International Conference on, IEEE, 2015, pp. 2414–2419.

[11] Chong, R.M., Tanaka, T., "Image extrema analysis and blur detection with identification," in Proceedings of IEEE international conference on Signal-Image Technology and Internet-based systems, pp. 320-326, 2008.

[12] Vladimir Vapnik, Statistical Learning Theory, Wiley, (1998).

[13] Roberto Battiti and Mauro Brunato, Statistical Learning Theory and Support Vector Machines (SVM) The LION way. Machine Learning plus Intelligent Optimization, Lionsolver Inc., Chapter 10, (2013).

[14] S. Das, "Comparison of various edge detection technique", International Journal of Signal Processing, Image Processing and Pattern Recognition, vol.9, no.2, (2016), pp.143-158.

[15] V. Saini and R. Garg, "A Comparative Analysis on Edge Detection Techniques Used in Image Processing", IOSR Journal of Electronics and Communication Engineering (IOSRJECE), ISSN : 2278- 2834, vol. 1, issue 2, (2012), pp. 56-59.

[16] R. C. Gonzalez, R. E. Woods and S. L. Eddins, "Digital Image Processing Using MATLAB", Pearson Education Ptd. Ltd, Singapore, (2004).

[17] Ce´sar Souza, Kernel Functions for Machine Learning Applications, http://crsouza.com/ (2010).

[18] Knerr, Stefan and Personnaz, Leon and Dreyfus, G´erard, ´SingleLayer Learning Revisited: A Stepwise Procedure for Building and Training a Neural Network, in Neurocomputing: Algorithms, Architectures and Applications, vol F68 of NATO ASI Series, (1990), 41–50.

[19] Friedman, Jerome H., Another approach to polychotomous classification, Department of Statistics, Stanford University, (1996).

[20] Krebel, Ulrich H.-G., Advances in Kernel Methods, Pairwise Classification and Support Vector Machines, MIT Press, (1999), 255–268.