Performance comparison of Support Vector Regression and Relevance Vector Regression for facial expression recognition

Gaurav Gupta
Student, Department of Electronics and Communication Engineering
Maharaja Surajmal Institute of Technology
Delhi, India
gauravg198@gmail.com

Neeru Rathee
Assistant Professor, Department of Electronics and Communication Engineering
Maharaja Surajmal Institute of Technology
Delhi, India
neeru1rathee@gmail.com

Abstract—This paper compares the performance of Relevance Vector Regression and Support Vector Regression for the purpose of facial expression recognition. The Support Vector Machine (SVM) is a state-of-the-art technique for regression and classification, but lacks the probabilistic treatment which is overcome by Relevance Vector Machine (RVM). Though SVM's have a good generalization performance, but their results are in general less sparse. This sometimes results in almost all of the training data to be used as Support Vectors. Comparing with RVM, the results obtained are relatively more sparse than SVM which results in lesser number of Relevance Vectors ultimately leading to lesser computation overhead. The above models are compared for facial expression recognition on Cohn Kanade database. Local Binary Pattern features are extracted from facial images. These are preprocessed for illumination and size, and also for dimensionality reduction before being used for training the RVM and SVM models. The paper concludes with a comparison of the SVM and RVM on the basis of test results.

Keywords—Relevance Vector Regression, Support Vector Regression, Facial Expression Recognition, Local Binary Pattern, Regression, SVM(Support Vector Maching), RVM(Relevance Vector Maching)

I. INTRODUCTION

Facial expression recognition has drawn significant interest and attention of the research community over the past few decades. Though literature related to expression recognition had been already present before 19th century [1], however, no real expression recognition systems appeared until the late 19th century as can be seen in [2]. Advances in computing and the development and implementation of machine learning algorithms made it possible to realize such systems. Today, expression recognition finds its applications in widely varying fields ranging from marketing [3] to robotics and HCI.

The effectiveness of a face recognition algorithm depends mainly on the two factors, firstly the effectiveness of the machine learning algorithm used to learn the underlying model or the hypothesis that best approximates the original hypothesis and secondly the feature extraction algorithm used to extract features that represent an image in the numerical domain for processing by the learning algorithm.

Following paper discusses two machine learning algorithms Relevance Vector Machine (RVM) and Support Vector Machine (SVM), and their performance comparison for facial expression recognition. Most of research related to RVM and SVM compares the two for classification [4-6] purposes and significant amount work has been done in regression [7]. Here,an empirical evaluation of RVM and SVM for regression based facial expression recognition is presented. Local Binary Patterns (LBP) were employedin the feature extraction process because of the ease of computation and it has already been successfully applied to expression and face recognition problems [19,20].

A brief summary of the rest of this paper is as follows: Section II covers the theoretical and mathematical background of SVR and RVR. Section III describes the LBP features. Next, Section IV and V discuss the steps performed, parameters used and the results obtained for RVR and SVR. Finally, Section VI concludes the paper with the comparison of RVM and SVM on the basis of results obtained in section V.

II. LEARNING ALGORITHMS

Both SVM and RVM are supervised learning methods where we are given a set of input vectors $\{ x_n \}_{n=1}^{N}$ along with the corresponding targets $\{ t_n \}_{n=1}^{N}$ and we wish to infer the underlying function *f(x)* in order to make accurate predictions of $\boldsymbol{t}$ for previously unseen values of $\boldsymbol{x}$. The function *f(x)* maps each input vector x_n to the target $t_n$which may be class labels (in classification) or real values (in regression).

A. Support Vector Regression

The Support Vector Machine pioneered by Vladimir N. Vapnik and Alexey Ya. Chervonenkis in 1963 was a binary linear classifier. Later in 1992 Vapnik and his coworkers proposed a nonlinear version of SVM [8] by employing the kernel trick [9].

The linear SVM works by constructing a hyper plane that separates the two classes of input vectors, while achieving maximum separation between those classes.

978-1-4673-6792-9/15/$31.00©2015 IEEE

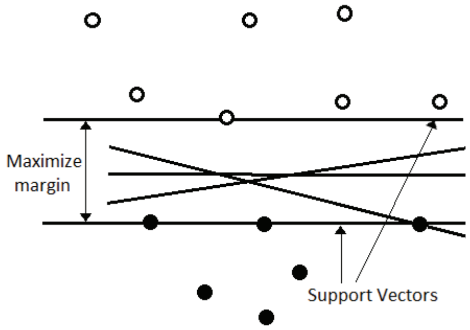

Fig. 1. Linear SVM model.

SVM's achieves non-linearity by using a non-linear mapping function *K(**x**,**x**)* that transforms the input ***x*** into an N dimensional non-linear output. Now a linear model can be constructed in this new feature space. The linear model in this feature space is given by:

$$f(\boldsymbol{x},\boldsymbol{w}) = \sum_{n=1}^{N} w_n . K(\boldsymbol{x},x_n) + w_0$$

Here w_0 represents the bias term and $K(x,x_n)$, $n = 1,2 \ldots N$ represents a set of non-linear transformations. Support Vector Regression uses a loss function *L (t, f (x, w))* to measure the quality of the estimation. SVR uses a loss function called ε-insensitive loss function [10].

$$L(\boldsymbol{t}, f(\boldsymbol{x},\boldsymbol{w})) = \begin{cases} 0 & , |\boldsymbol{t} - f(\boldsymbol{x},\boldsymbol{w})| - \varepsilon \leq 0 \\ |\boldsymbol{t} - f(\boldsymbol{x},\boldsymbol{w})| - \varepsilon , & otherwise \end{cases}$$

SVM uses ε-insensitive loss to perform linear regression in the high-dimension feature space while reducing model complexity by minimizing □*w*□^{2}.This is done by introducing slack variables ξ_n, ξ_n^*, $n = 1, 2, \ldots N$ to measure the deviation of training samples outside the ε-sensitive zone. So SVR is formulated as minimization of the following function:

$$min \ \frac{1}{2}\boldsymbol{w}^2 + C\sum_{n=1}^{N}(\xi_n + \xi_n^*) \qquad (1)$$

Subject to conditions:

$$\begin{cases} t_n - f(x_n,\boldsymbol{w}) \leq \varepsilon + \xi_n^* \\ f(x_n,\boldsymbol{w}) - t_n \leq \varepsilon + \xi_n \\ \xi_n, \xi_n^* \geq 0, n = 1, \ldots, N \end{cases}$$

The constant $C > 0$ determines the tradeoff between the flatness of *f* and the values up to which deviations greater than *ε* are tolerated. The solution to (1) is obtained by transforming it into a dual optimization problem. Finally the regression function is stated as:

$$f(\boldsymbol{x}) = \sum_{n=1}^{N}(\alpha_n^* - \alpha_n)K(\boldsymbol{x},x_n)$$

Where the kernel function

$$K(\boldsymbol{x},x_n) = \sum_{i=1}^{N} g_i(\boldsymbol{x})g_i(x_n) \qquad (2)$$

does the non-linear mapping of the linear input space to non-linear output space.

B. Relevance Vector Regression

Michael E. Tipping proposed the Relevance Vector Machine in 2000 [11] at Microsoft Research. RVM uses Bayesian inference to provide sparse solutions to regression and probabilistic classification problems [12].

RVM models the outputs t_n using a function $f(x_n,\boldsymbol{w})$ and some additive noise ε_n.

$$t_n = f(x_n,\boldsymbol{w}) + \varepsilon_n$$

where $f(x_n,\boldsymbol{w})$is the inference function and ε_n is assumed to zero mean Gaussian noise with variance σ^2 .

$$f(x,\boldsymbol{w}) = \sum_{i=1}^{N} w_i K(x,x_i)$$

$K(x,x_i)$is the kernel function as in (2).

RVM assumes the likelihood of the dataset to be Gaussian with variance $\boldsymbol{\sigma}^2$and mean *f (**x**)*. Instead of performing the maximum likelihood estimation of the parameters ***w*** and $\boldsymbol{\sigma}^2$RVR proceeds by introducing a prior $\boldsymbol{\mu}$ over the weights ***w*** which is also assumed to be zero mean Gaussian prior. $\boldsymbol{\mu}$ is a vector of N + 1 hyperparameters one for each weight. This parameter is responsible for the sparsity of RVR. RVR uses *automatic relevance determination* (ARD) priors and defines some hyperpriors over the priors $\boldsymbol{\mu}$ and the noise variance $\boldsymbol{\sigma}^2$. During the training phase, both the priors are re-estimated iteratively till a convergence criterion is met, during which many of the μ_n tend to infinity leading to many of the posterior probabilities of the associated weights tending to zero.

At the end of the iterative re-estimation procedure, the set of samples for which the inferred weights are non-zero are used for the estimation of new testing data. These sample points are termed as Relevance Vectors.

III. LOCAL BINARY PATTERN (LBP)

The Local Binary Pattern (LBP) operator was proposed by Timo Ojala in [13] as an extension to the texture unit model proposed by Wang and He [14]. This LBP operator was a simple 3 X 3 pixel neighborhood operator

p_0	p_1	p_2
p_7	p_c	p_3
p_6	p_5	p_4

Fig. 2. A 3X3 pixel neighborhood.

$$LBP(p_c) = \sum_{i=0}^{7} I(p_i - p_c)\, 2^i$$

where $p_i$denotes a pixel from the 3X3 neighbor of the central pixel p_c. $I(p_i - p_c)$ denotes the thresholding function which converts each neighboring pixel intensity to a binary symbol.

$$I(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

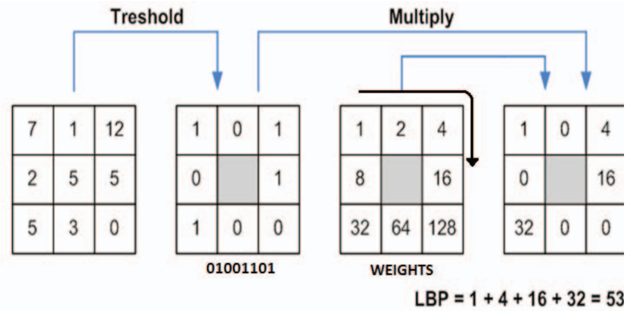

Fig. 3. Example working of LBP operator.

The above example shows how local binary patterns are obtained.The original LBP operator was later extended [15] as $LBP_{P,\ R}$. The parameter R denotes the radius of the operator, $R=1,2\ldots$ and P denotes the number of sampling points used from the neighborhood.The values of R determine spatial resolution and that of P control the angular resolution.

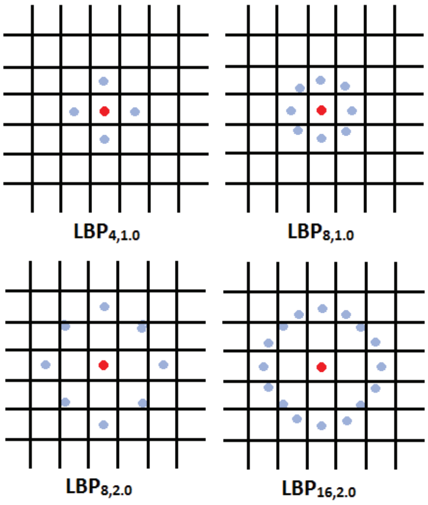

Fig. 4. $LBP_{P,R}$ for different (P,R) value pairs.

In case the sampling points lie on the edge of two pixel values than some interpolation technique can be employed to obtain the pixel value at that point.

IV. EXPERIMENTAL SETTING

A. Dataset

Images for testing and training were taken from the Cohn-Kanade database version 2, referred to as CK+[16].The CK+ is an extended version of the original CK database and contains 22% more sequences which includes 27% more subjects. CK+ consists of both posed (non-spontaneous) and non-posed expressions. Each sequence has fully FACS [17] coded target expressions.

For the tests a total of 1018 images was usedwhich included seven different labels anger, disgust, fear, neutral, happy, sad and surprise. These images were split up into training and testing image sets. For each subject exactly three images from the dataset were taken for each expression.

B. Preprocessing for size and illumination

Changes in illumination and image resolution can greatly affect the performance of expression recognition systems, so these effects need to be handled efficiently.

Resolution variations are taken care of by scaling all facial images cropped from the complete image to a fixed size.

Illumination changes between images are handled by contrast stretching (normalization) the facial images to the full range of pixel values.However, this method doesn't compensate for illumination variations appearing within an image due to non-uniform lighting conditions.

C. Expression Labels

Our experiments were based on the classification of expressions proposed by Paul Ekman in [18] which states the existence of six basic universal expressions, namely anger, disgust, fear, happy, sad and surprise. In addition to the six universal expressions we also use another class label 'neutral'.So a total of seven labels are used. Each label is assigned a real valuefrom -3 to +3.

D. LBP parameters

The LBP operator can be used with different values of the parameters R and P. In our experimental setup we have used R= 1 and P = 8. The values $R = 1, 2,$ 3 were also tested. Best results were obtained with R = 1. So all further results were taken with $LBP_{1,\ 8}$. Though with the values of R greater than 1, the value of P can also be more than 8, still R = 8 was used because of computational simplicity, leading to faster processing.

E. Feature Vector

To represent our facial image containing expressions, in the numeric domain the extracted face image after application ofLBP operator is divided into an 8X8 matrix of sub images. After the above step a 32-bin histogram of each sub image is taken as done in [13]. This step gives a numeric vector for each sub-image. These vectors are concatenated row wise in terms of image blocks to give a feature vector representation of an image.

Since not all portions of the facial image (as shown in the first image in Fig. 5) contain information. Hence histogram features of some blocks are skipped and are not added in the feature vector. These blocks are shown as darkened blocks in the second image in Fig. 5.

This technique helps in reducing dimensionality of the feature vector and at the same time it also removes some amount of redundant information which may be added by these sections of the image.

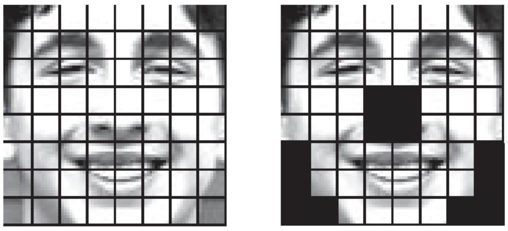

Fig. 5. Division of image into blocks

The images shown in Fig. 5 are not LBP applied images. These are images obtained directly after face extraction and have been used for convenience in understanding.

F. Training and Testing

Feature vectors of images created by the method described in the previous section and are used for training RVR and SVR models.Training and testing sets are made for performance evaluation as will be described in Section V. Various parameters associated with the training and testing part are:

*1) Kernel Function:*Radial Basis Function (RBF) kernel was used for both RVR and SVR.

$$K(x_i, x_j) = e^{-\frac{x_i - x_j^2}{2\sigma^2}}$$

where σ denotes the width of the Gaussian.

2) Dimensionality Reduction: Since the dimension of the input data is quite high, so a dimension reduction technique had to be used for faster processing and efficient representation of features. Principal Component Analysis (PCA) was used for dimension reduction.

3) Cross Validation Procedure: For the performance evaluation of both RVR and SVR we used subject-independent cross-validation procedure. Subject-independent cross-validation means that testing was done for images of subjects whose images were not used during training as was done by Mihalis in [19]. A 10-fold leave one out cross-validation(i.e. 90% of the sampleswere used for training and the remaining 10% were used for testing), was performed 10 times over the complete dataset for both SVR and RVR to obtain generalized performance.

4) Image Resolution: Resolution of facial images also affects the performance of our LBP based setup. Before feature extraction all images are resized to a fixed resolution. Results at many different resolutions were taken, however, only the resolutions that gave best results have been used for comparison.

G. Performance Criterion: Following measures has been used for evaluating performance of SVR and RVRmethods:

*1) Mean Square Error (MSE):*MSE is a risk metric that determines the expected value of squared error or loss.

*2) Squared Correlation Coefficient (Corr):*It determines the degree to which variables are linearly related. So a higher value is desirable. Squared Correlation Coefficient has also been adopted as a measure of performance since, sometimes mean square error can give misleading results for regression. This happens because MSE for some outliers may severely degrade the value of overall MSE.

3) Sparsity: The number of non-zero parameters determine the sparsity of a machine learning algorithm. Lesser is the number of non-zero elements, the higher is the sparsity and hence better is the system. Here the number of Support Vectors (SV's) and number of Relevance Vectors (RV's) determine sparsity. Therefor the number of Relevance Vectors (RVR) and Support Vectors (SVR) were also used for performance evaluation.

V. RESULTS

The results for both SVR and RVR are tabulated after tuning their respective parameters (σ, ε and C for SVR; and σ only for RVR) for best results.

Initial results shown in table II were taken at imagesscaled to the following resolutions: 70X70, 80X80 and 90X90.

Table 1. Results at Different Image Resolutions

Image Resolution	**SVR**			**RVR**		
	MSE	***Corr***	***SV's***	***MSE***	***Corr***	***RV's***
70X70	1.1636	0.7594	788.9	0.8814	0.7498	401.1
80X80	1.1039	0.7702	773.5	0.6694	0.7587	296.6
90X90	1.1824	0.7518	762.2	0.9550	0.7491	302.4

As can be inferred from the results, the resolution of 80X80 gave results better than others for both SVR and RVR. The values of *MSE* and *Corr* depict the effect of changing resolution; especially noticeable in the case of RVR. Hence, for further results 80X80 resolution was used.

As discussed in previous sections, PCA was used for dimensionality reduction of our feature vectors. So, both models were trained at varying ranges of number of principal components. Best results were obtained for the components from 2 to 345 for RVR and components from 2 to 245 for SVR. Results after application of PCA are tabulated in table III.

Table 2. Results After using PCA

Image Resolution	SVR			RVR		
	MSE	*Corr*	*SV's*	*MSE*	*Corr*	*RV's*
80X80	1.0389	0.7801	741.0	0.6074	0.7714	324.9

Results in table III show that in terms of MSE RVR clearly outperforms SVR, however the values of *Corr* show that SVR has performedslightly better than RVR.In terms of sparsity, the number of Support Vectors is nearly twice the number of Relevance Vectors.

Fig. 6 shows a plot of the actual curve and the regression curves for both RVR and SVR for a set of testing data chosen from one of the cross validation test sets.

VI. CONCLUSION

In this paper, we have compared two supervised learning techniques, namely SVM and RVM for facial expression recognition. RVM being relatively new, has not been explored much in comparison to SVM especially for regression tasks. Our work specifically compares SVM and RVM for regression only and evaluates them on the basis of sparsity of the learned model and their accuracy. On the basis of results we can conclude that RVM produces a much sparse model than SVM (nearly half of the Relevance Vectors as compared to Support Vectors). However the accuracy of both SVM and RVM is nearly the same. These results are in accordance with the results obtained in previous works comparing SVM and RVM for classification and regression.

Though the results observed through our system are sufficient to conclude about the performance of SVM and RVM, yet as a future work we can improve our feature extraction procedures to enhance the accuracy of the overall system. We can also use a feature extraction procedure other than LBP for performance evaluation. The work done in this paper uses images from the Cohn Kanade Database (version 2), so a generalization to other databases can also be done as an improvement to the current setup.

REFERENCES

[1] Charles Darwin, "The expression of the emotions in man and animals,"London: John Murray, 1872.

[2] A. Samal and P.A. Iyengar, "Automatic Recognition and Analysis of Human Faces and Facial Expressions: A Survey," Pattern Recognition, vol. 25, no. 1, pp. 65-77, 1992.

[3] Gurvinder Singh Shergill, Abdolhossein Sarrafzadeh, Olaf Diegel, Aruna Shekar, "Computerized Sales Assistants: The Application of Computer Technology to Measure Consumer Interest – A Conceptual Framework,"Journal of Electronic Commerce Research, VOL 9, NO 2,, pp. 176-191, 2008.

Xu Xiang-min, Mao Yun-feng, Xiong Jia-ni, Zhou Feng-le, "Classification Performance Comparison between RVM and SVM," IEEE International Workshop on Anti-counterfeiting, Security, Identification, pp. 208 – 211, , 16-18 April 2007.

[4] D.Datcu, L.J.M.Rothkr, "Facial Expression Recognition with Relevance Vector Machines," IEEE International Conference on Multimedia and Expo, pp. 193 – 196, July 2005.

[5] B. Demir,S.Erturk, "Hyperspectral Image Classification Using Relevance Vector Machines," IEEE Geoscience and Remote Sensing Letters, vol. 4, Issue: 4, pp. 586 – 590, Oct. 2007 .

[6] Mihalis A. Nicolaoua, Hatice Gunesb, Maja Pantica, "Output-associative RVM regression for dimensional and continuous emotion predictio," Image and Vision Computing, vol. 30, Issue 3,pp. 186–196, March 2012.

[7] Boser, B. E., Guyon, I. M., Vapnik, V. N. , "A training algorithm for optimal margin classifiers," Proceedings of the fifth annual workshop on Computational learning theory, pp. 144-152,1992.

[8] Aizerman, M., Braverman, E., and Rozonoer, L. "Theoretical foundations of the potential function method in pattern recognition learning," Automation and Remote Control, 25,p.p. 821–837 1964.

[9] H. Drucker, C. Burges, L. Kaufman, A. Smola, V. Vapnik, "Linear support vector regression machines,"Advances in Neural Information Processing Systems 9, 1997.

[10] Michael E. Tipping, "TheRelevanceVectorMachine," SaraA. Solla,ToddK. Leen,KlausRobertMuller,editors,Advances in Neural Information Processing Systems 12, Cambridge,MITPress, p.p. 652-658, 2000.

[11] Michael E. Tipping, "Sparse Bayesian Learning and the Relevance Vector Machine," Journal of Machine Learning Research 1, p.p. 211–244, 2001.

[12] Timo Ojala, Matti Pietikainen T, David Harwood, "A comparative study of texture measures with classification based on feature distributions," Pattern Recognition, vol. 29, p.p. 51-59, 1996.

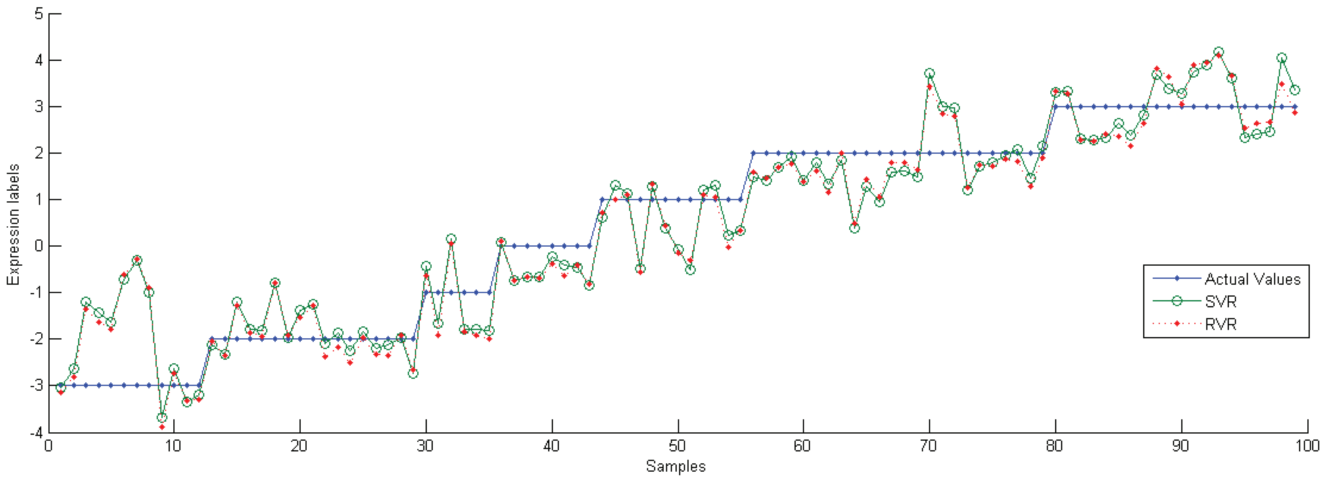

Fig 6. RVR and SVR results with actual values

[13] L. Wang and D. C. He, “Texture classification using texture spectrum,” Pattern Recognition 23, p.p. 905-910, 1990.

[14] Timo Ojala, Matti Pietikainen, Topi Maenpaa, “Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns,”IEEE transactions on pattern analysis and machine intelligence, vol. 24, no. 7, p.p. 971-987, July 2002.

[15] T. Kanade, J. Cohn, and Y. Tian, “Comprehensive database for facialexpression analysis,” IEEE International Conference on Face and Gesture Recognition, pp. 46–53, Mar. 2000.

[16] P. Ekman, W. Friesen, “Facial Action Coding System: A Technique for Measurement of Facial Movement,” Consulting Psychologists Press, 1978.

[17] Paul Ekman, Wallace V. Friesen, Phoebe Ellsworth, “Emotion In The Human Face: Guidelines for Research and an Integration of Findings,” Pergamon Press Inc., 1972.

[18] Caifeng Shan, Shaogang Gong, Peter W. McOwan, ”Facial expression recognition based on Local Binary Patterns: A comprehensive study,” Image and Vision Computing 27, p.p. 803–816, 2009.

[19] Ahonen, Hadid, Pietikainen, “Face Description with Local Binary Patterns: Application to Face Recognition”, IEEE transactions on pattern analysis and machine intelligence, vol. 28, no. 12, Dec 2006.