Video-based facial discomfort analysis for infants

E. Fotiadou[a], S. Zinger*[a], W.E. Tjon a Ten[b], S. Bambang Oetomo [a,b], P.H.N. de With[a]
[a]Eindhoven University of Technology, Den Dolech 2, 5612 AZ, Eindhoven, The Netherlands;
[b]Maxima Medical Center, De Run 4600, 5504 DB, Veldhoven, The Netherlands

ABSTRACT

Prematurely born infants receive special care in the Neonatal Intensive Care Unit (NICU), where various physiological parameters, such as heart rate, oxygen saturation and temperature are continuously monitored. However, there is no system for monitoring and interpreting their facial expressions, the most prominent discomfort indicator. In this paper, we present an experimental video monitoring system for automatic discomfort detection in infants' faces based on the analysis of their facial expressions. The proposed system uses an Active Appearance Model (AAM) to robustly track both the global motion of the newborn's face, as well as its inner features. The system detects discomfort by employing the AAM representations of the face on a frame-by-frame basis, using a Support Vector Machine (SVM) classifier. Three contributions increase the performance of the system. First, we extract several histogram-based texture descriptors to improve the AAM appearance representations. Second, we fuse the outputs of various individual SVM classifiers, which are trained on features with complementary qualities. Third, we improve the temporal behavior and stability of the discomfort detection by applying an averaging filter to the classification outputs. Additionally, for a higher robustness, we explore the effect of applying different image pre-processing algorithms for correcting illumination conditions and for image enhancement to evaluate possible detection improvements. The proposed system is evaluated in 15 videos of 8 infants, yielding a 0.98 AUC performance. As a bonus, the system offers monitoring of the infant's expressions when it is left unattended and it additionally provides objective judgment of discomfort.

Keywords: video analysis, discomfort detection, pain, infant face detection, Active Appearance Model (AAM)

1. INTRODUCTION

Clinically, pain and discomfort are considered as subjective experiences and are typically measured by patient self-report. Healthy adults are able to indicate the intensity, location and duration of their pain. However, infants do not have the ability to communicate verbally and thus are unable to self-report. Inability to provide a reliable report about pain leaves the neonates vulnerable to under-recognition, and under- or over-treatment. Nurses and parents are then responsible to assess the discomfort of the infant and take action whenever treatment is needed.

The assessment of pain in infants is considered as one of the most challenging problems in neonatology. There are many reasons for recognizing pain in infants. Most significantly, pain is a major indication of infant illness. Furthermore, the quality of the care that an infant receives depends largely on the quality of the pain assessment. It is important to recognize and treat pain, since persistent unrelieved pain can cause severe complications, such as nervous system changes and delayed development[1,2]. Despite the significance of pain recognition, most neonatal intensive care units do not have sufficient resources for monitoring it continuously.

Infants feeling pain and distress experience behavioral and physiological changes[3,4]. Behavioral pain indicators include crying, changes in body movements and changes in facial expressions. Physiological changes involve increase in heart rate, respiratory rate, and blood pressure, as well as changes in the levels of oxygen and carbon dioxide in the blood. Many pain assessment tools have been created to assist healthcare professionals to identify and quantify pain and discomfort, such as the COMFORT[5], PIPP[6] (Premature Infant Pain Profile), BIIP[7] (Behavioral Indicators of Infant Pain) and MIPS[8] (Modified Infant Pain Scale). However, there is currently no broadly accepted tool to assess neonatal pain. The main controversy surrounding the use of such assessment tools is the subjectivity of the observer. Studies have demonstrated that health professionals are not entirely impartial in their judgments[9]. In addition, manual assessment of pain is time-consuming.

* s.zinger@tue.nl

Visual Information Processing and Communication V, edited by Amir Said, Onur G. Guleryuz, Robert L. Stevenson, Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 9029, 90290F
 · doi: 10.1117/12.2037661

Facial expressions play a major technical role in discomfort assessment, as they are the most specific and frequent discomfort indicators. This is also attested by the fact that most pain assessment tools rely mainly on the facial expressions of the infant. Although there is a vast potential for using computer vision to assess discomfort and pain, there are very few articles in the literature addressing this issue. Brahnam et al.[10] are the first to use various face classification techniques, including Principal Component Analysis (PCA), linear discriminant analysis, Support Vector Machines (SVM) and Neural Network Simultaneous Algorithm (NNSOA), to classify the facial expressions of 26 neonates into pain and no-pain classes. Prior to classification, the infant images are manually pre-processed (cropped, rotated and scaled), the faces are centered within an ellipse and then their dimensions are reduced by PCA. The experiments have demonstrated that such classification techniques can achieve a reasonable accuracy. NNSOA provides the best classification rate of pain versus non-pain (90.20%), followed by SVM with a linear kernel (82.35%). However, this system is not fully automatic because it requires manual pre-processing, which will considerably limit its application in a hospital environment.

Lu et al.[11] has applied a 2D Gabor filter on infant facial images to extract expression features. The images are manually segmented, rotated, scaled and their contrast is enhanced. Adaboost is then used for removing redundant Gabor features. For the classification, SVM is selected achieving 85.29% pain recognition rate in 510 facial images of 57 neonates. Again, this approach is only semi-automatic. In the pilot system designed by Han et al.[12], important facial features, such as eyes, mouth and eyebrows, are automatically extracted and analyzed for the purpose of discomfort detection. To further adapt this system to a real hospital setting, non-ideal situations such as changes in lighting conditions and viewpoint are taken into consideration. For the evaluation of their system, video recordings of one healthy newborn with different conditions are used, achieving 88% accuracy. For successful detection, this system requires good visibility of several facial features, such as eyes, mouth and eyebrows which is not always the case in the observation video. Lucey et al.[13] explore the UNBC-McMaster Shoulder Pain Archive to classify video sequences of adults as pain and no-pain. An AAM is employed to track the face and derive features by decoupling the face into rigid and non-rigid shape and appearance parameters. An SVM classifier with linear kernel is then applied for pain classification. In their work, they find that the fusion of all AAM representations produces higher recognition rate, revealing that these representations contain complementary information. The area under ROC curve that is finally achieved is 0.847.

Our work adopts the approach of Lucey et al.[13] for the problem of neonatal discomfort detection, but extends it in the following aspects. First, a robust initialization and recovery technique is incorporated to the AAM face tracker, providing face recovery after partial or total face occlusion that may occur due to infant movements or external objects. Instead of using the coarse AAM appearance representations for classification, descriptors of high discriminative power are extracted to boost the performance of the system. Furthermore, to deal with temporal aspects of discomfort detection, an averaging filter is applied to the classification outputs to stabilize detection results. Experiments with different image pre-processing methods are also conducted to evaluate possible recognition performance improvements.

The rest of the paper is organized as follows. Section 2 presents the proposed algorithm for automatic discomfort detection. Experimental results and evaluation of the proposed method are provided in Section 3. Finally, conclusions are drawn in Section 4.

2. DISCOMFORT DETECTION SYSTEM

An overview of our system is given in Figure 1. In the proposed system, an AAM is employed to track the face and extract visual features. Histogram-based texture features are then extracted from the AAM appearance representations to boost the algorithm's performance. SVMs are used to classify discomfort, while the outputs of SVM classifiers trained on different features are fused together via Logistic Linear Regression (LLR) to obtain a higher detection accuracy. Finally, the SVM output scores of successive frames are averaged. The following subsections describe the components of the system analytically.

2.1 Face tracking

2.1.1 Active Appearance Models (AAM)

AAM are non-linear, generative models, capable of synthesizing images of a given object class. The modeled object properties are the shape and the appearance of the object. The shape of an AAM[14] is described by a number of landmark

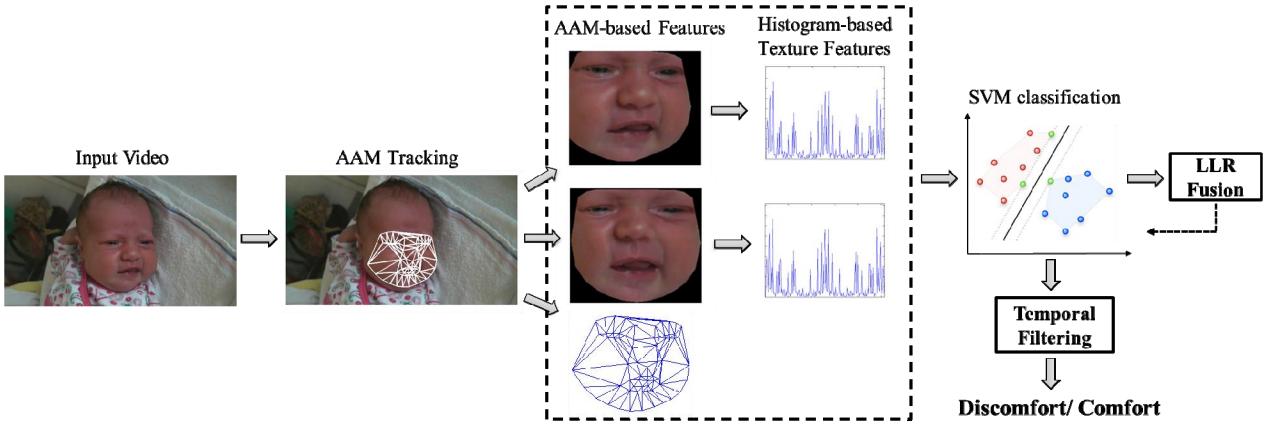

Figure 1. Block diagram of the proposed system. The face is tracked with an AAM and we obtain shape and appearance features from face tracking results. Histogram-based texture features are extracted from the AAM appearance representations. The features are used to classify infant discomfort with an SVM classifier. The SVM scores of individual classifiers are fused together with LLR. Finally, the SVM scores of successive frames are averaged.

points collected in a vector $\mathbf{s}, \mathbf{s} = (x_1, y_1, x_2, y_2, \ldots, x_u, y_u)^T$, that define a mesh, where u is the number of the landmark points. For building an AAM, a set of images, together with a set of landmark points, is needed. Following the data acquisition, for the purpose of achieving statistical validity, all shapes are represented on the same referential using Procrustes Analysis[15]. As a final step in the construction of the shape model, Principal Component Analysis[14] (PCA) is applied to the aligned shapes. A shape instance can then be expressed as a mean shape $\mathbf{s}_0$ plus a linear combination of n shape vectors $\mathbf{s}_i$, which results in

$$\mathbf{s} = \mathbf{s}_0 + \sum_{i=1}^{n} p_i \mathbf{s}_i , \tag{1}$$

with p_i being the shape parameters. The AAM appearance is represented by the number of pixels lying inside the mean shape $\mathbf{s}_0$. Similar to shape, the appearance $\mathbf{A}$ is represented as a mean appearance $\mathbf{A}_0$ plus a linear combination of m appearance vectors $\mathbf{A}_i$:

$$\mathbf{A} = \mathbf{A}_0 + \sum_{i=1}^{m} \lambda_i \mathbf{A}_i , \tag{2}$$

where λ_i are the appearance parameters. The fitting of the AAM model to an image is performed by specifying the shape and the appearance parameters, p_i and λ_i, so that the model instance is as close as possible to the target image.

2.1.2 AAM-based face tracking

In the proposed system, an AAM is used to track the infant's face through the frames of a video. AAM-based tracking has the advantages of accurate alignment, high efficiency and effectiveness in handling face deformation. The face tracking is performed by fitting an AAM in each video frame by using the inverse compositional algorithm[16,17].

The AAM fitting algorithm requires a suitable initial estimation of the face shape and position to find a proper landmark matching. In the first video frame, the AAM initialization is performed as follows. Firstly, the exact position of the face is detected in the video frame by a face detector based on a skin-color Gaussian Mixture Model (GMM). Secondly, inside the face region, the positions of the eyes and mouth are detected by generating eye and mouth candidates based on the gradient information of the face. Facial feature geometrical verification of the potential candidates follows to determine suitable eye-and-mouth pairs. Then, the pose of the face is coarsely estimated in order to select one out of three facial shapes: a left, a frontal and a right shape. For the pose estimation, the ratio of the distances between each eye

and the face border is calculated and a decision is made after comparison with an empirical threshold. The final position of the initial shape is obtained after rotation, translation and scaling of the shape, according to the estimated eye and mouth positions. An example of the overall AAM initialization procedure is shown in Figure 2.

Unlike the first frame, in the remaining frames, sufficient shape information is provided by the previous frames. The estimated facial shape of each frame can be used to initialize the AAM in the next frame. However, when the fitting error is higher than a predefined threshold, which means that the tracking is lost, then tracking recovery is employed by providing an initial AAM estimate, as it is performed for the first frame. Thus, the algorithm is able to recover the face in cases where the tracking is lost due to occlusions, obstructions or sudden movements.

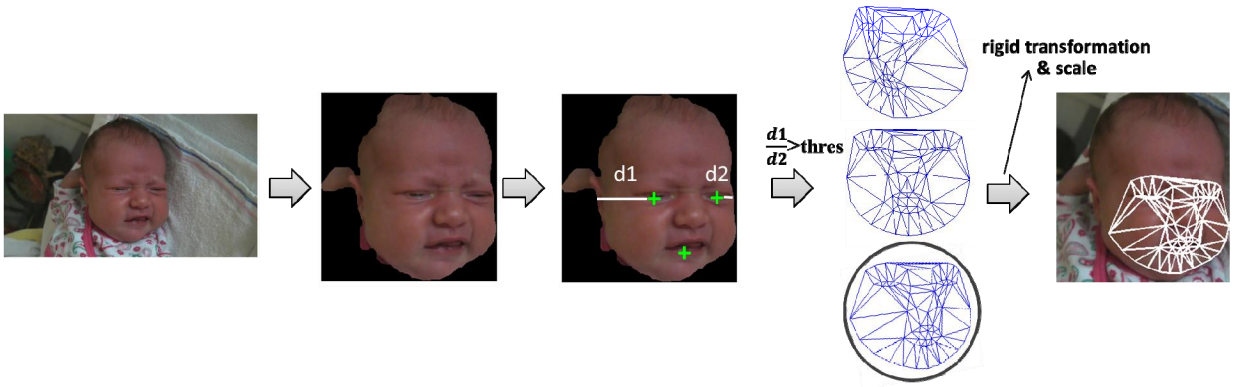

Figure 2. AAM initialization example. The face is detected in the video frame based on skin color. The positions of the eyes and mouth are located in the face based on gradient information and geometrical relations of the eye-mouth pair. The pose of the face is coarsely estimated to select a suitable facial shape. The facial shape is then transformed to obtain the initial shape.

2.2 Feature extraction

2.2.1 AAM-based features

Once the face is tracked with an AAM by estimating the shape and the appearance parameters, facial features can be obtained based on this information. These features can be later used for efficient discomfort classification. According to Lucey[13], the following three features can be extracted based on AAM parameters.

- SPTS (similarity-normalized shape): This is a vector in the form of the shape vector $\mathbf{s}$ which contains the coordinates of the landmark locations after all rigid geometric variations (translation and rotation) and scale are removed with respect to the mean shape $\mathbf{s}_0$.
- SAPP (similarity-normalized appearance): SAPP refers to the appearance after the removal of the rigid geometric variations and scale. This is achieved by warping the pixels of the facial image in the similarity-normalized shape.
- CAPP (canonical-normalized appearance): CAPP is the appearance when all non-rigid shape variation has been removed with respect to the mean shape $\mathbf{s}_0$. This is accomplished by warping the appearance of the facial image into the mean shape.

An example of the AAM-based features is given in Figure 3.

2.2.2 Histogram-based texture features

To improve the performance of the system, highly discriminative histogram-based texture descriptors are extracted from the AAM appearance representations. The following descriptors are extracted and compared: Local Binary Patterns (LBP), Elongated Local Binary Patterns (ELBP), Completed Local Binary Patterns (CLBP), Completed Elongated Local Binary Patterns (CELBP), Local Phase Quantization (LPQ) and Histogram of Oriented Gradients (HOG). These descriptors have shown high performance in facial expression recognition tasks, while their dimensionality and computational time are low, allowing for real-time applications.

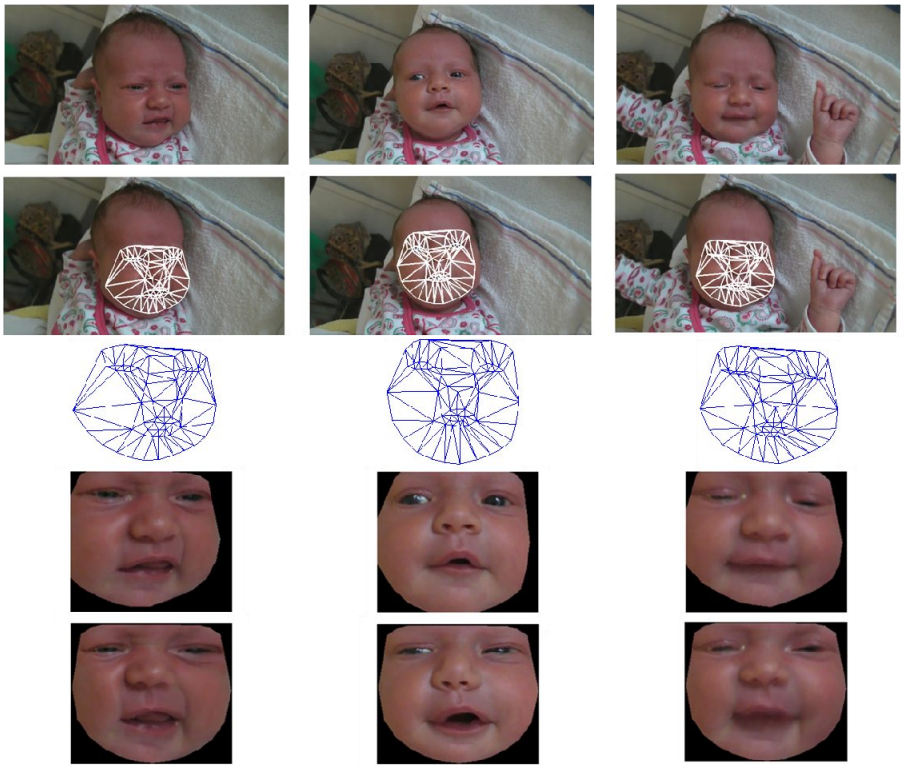

Figure 3. Example of the output of AAM tracking and the associated AAM-based features. Top row: original video frames; Second row: AAM-tracked frames; Third row: similarity-normalized shape features (SPTS); Fourth row: similarity-normalized appearance features (SAPP); Bottom row: canonical-normalized appearance features (CAPP).

The descriptors are extracted both on similarity-normalized appearance (SAPP) images and canonical-normalized appearance (CAPP) images for comparison. In the sequel of the paper, we denote an extracted descriptor on SAPP image as "<name>_SAPP", e.g. "LBP_SAPP", and on CAPP image as "<name>_CAPP", e.g. "LBP_CAPP". Prior to the feature extraction, the image is divided into blocks and the computed histograms of each block are concatenated. A brief description of each descriptor is given below.

1) *Local Binary Pattern* (LBP): LBP[18] is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considering the result as a binary number. In a grayscale image, the LBP label, for a pixel $\mathbf{x}$ with gray value g_c, is calculated by comparing g_c with the values g_i, of its P neighboring pixels at distance R (these pixels are located on a circle of radius R):

$$LBP^{P,R}(\mathbf{x}) = \sum_{i=1}^{P} d(g_i - g_c)2^{i-1}, \tag{3}$$

where $d(z)$ is defined as:

$$d(z) = \begin{cases} 1, & if\ z \geq 0, \\ 0, & if\ z < 0. \end{cases} \tag{4}$$

Ojala et al.[18] have shown that a small subset of the patterns accounted for the majority of the textures of images. These are called uniform patterns and they contain at most two bit-wise transitions from 0 to 1 for a circular binary string. By using only uniform patterns, the dimensionality of the LBP descriptor is significantly decreased.

An Elongated Local Binary Pattern[19] (ELBP) is a variant implementation of the conventional LBP which uses an elliptic, instead of circular neighborhood definition. ELBP has shown better performance in applications where the images contain anisotropic structures, such as face recognition.

2) *Completed Local Binary Pattern* (CLBP): The LBP descriptor considers only sign parameters and thus some textural information may be lost. This observation has motivated Zhenhua et al.[20] to extend it to the CLBP descriptor, by considering not only sign, but also magnitude and the gray value of the central pixel. Here we use only the sign, CLBP_S, and the magnitude, CLBP_M, of the CLBP descriptor. The CLBP_S operator is identical to the LBP operator. The CLBP_M operator is defined by:

$$CLBP\_M^{P,R}(\mathbf{x}) = \sum_{i=1}^{P} t(g_i - g_c, c)2^{i-1}, \tag{5}$$

where g_i is the intensity value of the i-th of the P neighbors at distance R, g_c is the gray value of the central pixel, c is a threshold to be determined adaptively (here it is set as the mean value of $(g_i - g_c)$ from the whole image) and $t(z,c)$ is defined as:

$$t(z,c) = \begin{cases} 1, & if\ z \geq c, \\ 0, & if\ z < c. \end{cases} \tag{6}$$

CLBP_S and CLBP_M are combined by concatenating their histograms to form the final descriptor. Furthermore, we examine the use of elliptic neighborhood both for CLBP_S and CLBP_M, like in ELBP, and we name the resulting descriptor Completed Elongated Binary Pattern (CELBP).

3) *Local Phase Quantization* (LPQ): The LPQ operator has originally been proposed by Ojansivu and Heikkila[21] as a texture descriptor that is robust to image blurring. LPQ employs the local phase information extracted using the 2-D Short-Time Fourier Transform (STFT) computed over a rectangular $M \times M$ neighborhood N_x at each pixel position $\mathbf{x}$ of the image $I(\mathbf{x})$. The STFT is defined as follows:

$$F(\mathbf{u}, \mathbf{x}) = \sum_{\mathbf{y} \in N_x} I(\mathbf{x} - \mathbf{y}) e^{-j2\pi \mathbf{u}^T \mathbf{y}}. \tag{7}$$

The local Fourier coefficients are computed at four frequency points, $\mathbf{u}_1 = [\alpha, 0]^T$, $\mathbf{u}_2 = [0, \alpha]^T$, $\mathbf{u}_3 = [\alpha, \alpha]^T$ and $\mathbf{u}_4 = [\alpha, -\alpha]^T$, where α is a sufficiently small scalar. For each pixel position, this results in a vector $\mathbf{F}_x = [F(\mathbf{u}_1, \mathbf{x}), F(\mathbf{u}_2, \mathbf{x}), F(\mathbf{u}_3, \mathbf{x}), F(\mathbf{u}_4, \mathbf{x})]$. The phase information in the Fourier coefficients is recorded after simple observation of the signs of the real and imaginary parts of each component in $\mathbf{F}_x$. For this purpose, a simple two-level scalar quantizer is used:

$$q_j = \begin{cases} 1, & if\ \mathbf{G}_j \geq 0, \\ 0, & otherwise, \end{cases} \tag{8}$$

where $\mathbf{G}_j$ is the j-th component of the vector $\mathbf{G}_x = [\mathrm{Re}\{\mathbf{F}_x\}, \mathrm{Im}\{\mathbf{F}_x\}]$. The eight binary values that are obtained from the quantization procedure are then represented as integers in the interval $[0; 255]$ using binary coding:

$$LPQ(\mathbf{x}) = \sum_{j=1}^{8} q_j 2^{j-1}. \tag{9}$$

4) *Histogram of Oriented Gradients* (HOG): The HOG[22] descriptor has first been described by Dalal and Triggs[23]. The HOG is a descriptor which counts the occurrences of gradient orientation in localized portions of an image. The first step in a HOG calculation is the computation of the gradient values. Typically, a horizontal and a vertical gradient filter are applied to the image, specified by

$$\mathbf{h}_x = [-1 \quad 0 \quad 1] \quad \text{and} \quad \mathbf{h}_y = [-1 \quad 0 \quad 1]^T. \tag{10}$$

The second step involves the cell histogram creation. Each pixel within the cell casts a weighted vote for an orientation-based histogram channel. The shape of the cells can be either rectangular or radial and the histogram channels are evenly spread over 0 to 180 degrees or 0 to 360 degrees, depending on whether the gradient is unsigned or signed, respectively. The gradient strengths are then normalized by grouping the cells together into larger, spatially connected blocks. This is done in order to account for changes in illumination and contrast. The vector of the components of the normalized cell histograms, from all of the block regions, forms the HOG descriptor. The blocks usually overlap, meaning that each cell contributes more than once to the final feature vector.

2.2.3 Image pre-processing

Without any image pre-processing, the color AAM appearance representations are used while they are converted to grayscale, which is required for histogram-based appearance feature extraction. In our experiments, two pre-processing methods are also evaluated: illumination normalization[24] and Laplacian of Gaussian (LoG) filtering. As a situation-independent system is important, we focus on handling difficult illumination conditions. For this purpose, illumination normalization[24] is used to pre-process the images prior to the feature extraction and the classification. Illumination normalization is a three-step procedure that involves Gamma correction, difference of Gaussians filtering and contrast equalization. Illumination normalization significantly reduces the influence of illumination variations, local shadowing and highlights, while preserving the elements of visual appearance that are needed for recognition. Aiming at revealing more detailed information about the local orientations in the image, LoG filtering is also used for pre-processing. LoG smoothens the image using a Gaussian filter and then applies a Laplacian filter. Figure 4 presents the different pre-processing methods that are used in our experiments.

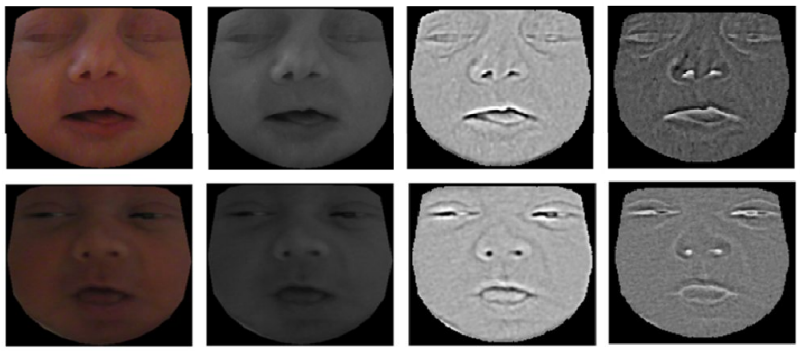

Figure 4. Two examples of the different image pre-processing methods used in our experiments. (a) Color images, (b) grayscale images, (c) illumination normalized images and (d) LoG filtered images.

2.3 Support Vector Machine (SVM) classification

SVM[25] is one of the most popular classifiers in facial expression recognition. Given a set of training examples, each marked as belonging to one of two classes, an SVM training algorithm builds a model that assigns new examples to one class or the other. Many hyperplanes can be found to separate these two classes. However, SVM aims to find the optimal separating hyperplane with the maximum margins that leads to the most effective classification. In case that the training data cannot be separated linearly, they are mapped to a high-dimensional space in which they are linearly separable. The mapping of the data can be achieved by using a kernel function. Here, a radial basis function (RBF) kernel is applied. The hyperplane is then calculated in the high-dimensional space. SVM is very sensitive to its parameters (kernel parameters, soft margin parameter) and thus careful parameter selection is of utmost importance.

2.4 Combination of individual SVMs

According to Lucey et al.[13], there is complementary quality in AAM feature representations and a higher detection accuracy can be achieved when these features are fused. Motivated by this, we examine the effect of fusing more

features together. Instead of fusing the features prior to the classification, the outputs of the classifiers trained on the different features are combined. For this purpose, Logistic Linear Regression[26] (LLR) is adopted, that calibrates the output scores of the individual SVM classifiers in a common domain in order to combine them effectively. Assume that we have N discomfort detectors with output scores $(o_1, o_2, \ldots, o_N)$. LLR calibrates all the individual scores through learning the weights $(a_0, a_1, \ldots, a_N)$ via logistic regression, so that the calibrated score CS is:

$$CS_N = a_0 + a_1 o_1 + a_2 o_2 + \ldots + a_N o_N , \tag{11}$$

where the constant a_0 improves the calibration through regularization. The weights are calculated by minimizing a cost function that is formed based on a set of training scores[26]. The BOSARIS Toolkit[27] was used for calibrating and fusing the various SVM scores.

3. EXPERIMENTAL RESULTS

3.1 Dataset

We have monitored the facial expressions of 10 infants in the neonatal intensive care unit of the Máxima Medical Center (MMC) in Veldhoven, The Netherlands. Informed consent is obtained from one or both parents of each infant. The videos are taken using a high-definition camera. The resolution of each frame is 1920x1080 pixels, while the frame rate is 25 fps. The duration of the videos varies from a few seconds to several minutes. The infants are monitored experiencing heel puncture, diaper change, hunger, resting or sleeping, under unconstrained lighting conditions. In Figure 5, there are some example frames of four videos, captured for four different infants.

Figure 5. Example frames from four infant videos. Each row corresponds to a video from a different infant.

3.2 Parameter selection

In order to train the discomfort classifier, important parameters should be first selected. These parameters are the parameters of the SVM classifier and the number of blocks for the histogram-based descriptors. Tuning the parameters of the SVM classifier controls the tradeoff between underfitting and overfitting. Furthermore, when using histogram-based appearance features, the image is divided into a number of blocks of predefined size (but varied in the optimization) both horizontally and vertically and the extracted histograms of the blocks are concatenated to form the final descriptor. The number of blocks per dimension is an important parameter, as we wish to increase the discriminative power of the descriptor by an efficient partitioning of the image. For the parameter selection procedure, 158 training images are chosen from the video recordings of the 10 neonates, half of them displaying discomfort and half of them comfort. The SVM parameters and the number of blocks per dimension are then selected such that they maximize the accuracy in the training set by using leave-one-subject-out cross-validation.

Furthermore, each histogram-based descriptor has its own parameters. For computational efficiency purposes, simple versions of each descriptor are selected. For LBP and CLBP, we select a neighborhood of 8 pixels at unity distance. For ELBP and CELBP, a neighborhood of 8 pixels at horizontal distance of 2 pixels and vertical 1 pixel is chosen. For these four descriptors, only uniform patterns are used. The resulting number of dimensions of LBP and ELBP descriptors is 59 for each block, while for CLBP and CELBP it is doubled because of the magnitude information. LPQ descriptors have always 256 dimensions for each block, independently of the window size. Thus, the window size for the LPQ calculation is set to 5, as it shows good performance according to our experiments. For HOG, the histogram channels are calculated over rectangular blocks by the computation of unsigned gradients. We use 9 rectangular blocks and 9-bin histograms per block, resulting in the feature vector containing 81 elements.

3.3 Performance evaluation

For the evaluation of the system, 15 videos of 8 infants are used, 5 displaying comfort, 1 discomfort and 9 videos exhibiting both. The videos of the remaining 2 infants are considered inappropriate, as most of the frames exhibited occlusion of the infant's face, due to moving hands or large face rotation, such that both eyes are not visible. As the overall number of frames displaying discomfort is quite different from the one with comfort, the accuracy cannot be used as a reliable metric. Instead, the Receiver Operator Characteristic (ROC) curve is used as a more reliable performance metric. The ROC curve is obtained by plotting the true positive rate against the false positive rate at various decision thresholds of the classifier. The area under ROC curve (AUC) is used to assess the performance of the system. The AUC metric ranges from 0.5 (pure chance) to 1.0 (ideal classification).

Due to the limited amount of data, our dataset cannot be divided into a training set and test set. Thus, leave-one-subject-out cross-validation is used to evaluate the performance of the system. According to this approach, the video frames of a single infant are used as validation data, while a subset of the training images that does not contain images of this subject is used to train the classifier. This procedure is repeated for all the subjects and the performance measurement is carried out by averaging the estimated AUC of all subjects.

3.3.1 AAM-based face tracking evaluation

The AAM tracking algorithm is tested in 15 infant videos. To improve tracking performance and robustness, prior to tracking, a person-specific[28] and grayscale AAM is constructed for each infant. The results of the AAM-based face tracking in the test videos are presented in Table 1. The infant face is correctly tracked in 37,829 out of 43,823 (86.3%) test frames. Figure 6 shows the typical tracking results of four videos (V_4, V_6, V_{11}, and V_{13} in Table 1).

The AAM loses the face during tracking mostly when the hands of the infants or blankets cause partial or total occlusion and in cases that the out-of-plane face rotation is such that both eyes are not visible. Figure 7 presents example video frames displaying such situations. However, when there is no occlusion, the position of the face is quickly recovered.

Table 1. Performance of the AAM tracking algorithm. Video sequences V_1 to V_{15} denote the 15 infant test videos. The second row of the table shows the total number of frames while the third row presents the frame count with lost tracking.

	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9	V_{10}	V_{11}	V_{12}	V_{13}	V_{14}	V_{15}	**sum**
#frames	938	3818	1186	1141	463	1090	4698	1929	2932	3553	1486	2065	5599	1933	10,992	**43,823**
#lost frames	30	45	23	5	14	231	1113	63	323	1615	447	232	1324	6	523	**5,994**

3.2.2 Discomfort detection evaluation

The discomfort detection performance when using individual features is illustrated in Tables 2 and 3. The estimated AUC for the various features and different image pre-processing methods is presented. The AUC is estimated both for the initial unprocessed SVM outputs, but also for the post-processed outputs with an average filter. The temporal depth of the averaging filter is set in such a way that it maximizes the AUC of the cross-validation. Only the frames that are correctly tracked are used for the performance estimation.

As we observe in Tables 2 and 3, in all cases, temporal filtering improves significantly the classification result. Sometimes, inaccuracies in the face tracking lead to false classification in a few frames. However, as neighboring frames

Figure 6. Face tracking results of videos V_4 (top row), V_6 (second row), V_{11} (third row) and V_{13} (bottom row) in Table 1.

Figure 7. Example video frames where face tracking is lost due to partial face occlusion or large out-of-plane face rotation.

usually belong to the same class, an average filter smoothes the result removing the false detections.

Regarding the image pre-processing methods, when AAM appearance representations are used (see Table 2), grayscale images lead to the highest performance, while LoG-filtered images cause the lowest performance. As we can see in Table 3, when histogram-based texture features are extracted, the performance of all image pre-processing methods is comparable. Here, we should underline the fact that even though the illumination conditions during the video recordings are uncontrolled, most of the test videos have similar illumination conditions. In case of videos with more illumination variations, the illumination-normalized images would probably yield better results. However, this is not yet confirmed by experiments.

Concerning the AAM representations, similarity-normalized shape (SPTS) yields the highest detection rate, as can be seen in Table 2. If we compare the similarity-normalized appearance (SAPP) and the canonical-normalized appearance (CAPP), CAPP achieves a better recognition rate. In Table 3, we observe that the use of SAPP and CAPP images exhibit similar performance, while the highest scores for grayscale, illumination-normalized and LoG-filtered images are achieved when using CAPP images. CAPP- based classification is expected to perform better compared to SAPP images, since in many frames the infants are not in frontal position, while good face registration is of high importance for all situations. However, when the face rotation is large, the facial features are highly distorted after the warping to the frontal pose. Furthermore, CAPP features are more sensitive to tracking inaccuracies, as the warping strongly depends on accurate facial feature point estimation.

For the individual features, the maximum area under ROC curve (AUC) is 0.972 and this result is obtained by Completed Local Binary Patterns (CLBP), computed on grayscale CAPP. Without temporal filtering, the highest AUC is 0.916 and is achieved by Elongated Local Binary Patterns (ELBP) computed on grayscale SAPP.

Table 2. Estimated AUC (without and with SVM output averaging) for AAM-based features and different image pre-processing methods. The SPTS feature is based on AAM landmark points and is therefore not influenced by pre-processing.

Feature	Pre-processing							
	None		Grayscale		Illumination Normalization		LoG filtering	
	AUC	AUC AVG filter	AUC	AUC AVG filter	AUC	AUC AVG filter	AUC	AUC AVG filter
SPTS	0.912	**0.958**	-	-	-	-	-	-
SAPP	0.832	0.896	0.858	0.919	0.764	0.804	0.771	0.832
CAPP	0.861	**0.921**	0.856	**0.939**	0.829	**0.914**	0.780	**0.875**

Table 3. Estimated AUC (without and with SVM output averaging) for histogram-based appearance features and different image pre-processing methods.

Feature	Pre-processing					
	Grayscale		Illumination Normalization		LoG filtering	
	AUC	AUC AVG filter	AUC	AUC AVG filter	AUC	AUC AVG filter
LBP_SAPP	0.892	0.941	0.896	0.945	0.865	0.920
ELBP_SAPP	0.916	0.965	0.898	0.956	0.873	0.924
CLBP_SAPP	0.889	0.940	0.900	0.949	0.834	0.890
CELBP_SAPP	0.905	0.954	0.898	0.949	0.906	0.952
LPQ_SAPP	0.898	0.949	0.872	0.921	0.856	0.910
HOG_SAPP	0.904	0.949	0.891	0.939	0.871	0.936
LBP_CAPP	0.901	0.962	0.889	0.951	0.889	0.950
ELBP_CAPP	0.898	0.964	0.883	0.948	0.891	0.949
CLBP_CAPP	0.914	**0.972**	0.898	0.950	0.895	0.950
CELBP_CAPP	0.906	0.960	0.911	0.966	0.881	0.940
LPQ_CAPP	0.893	0.949	0.915	0.961	0.897	0.957
HOG_CAPP	0.839	0.938	0.900	**0.970**	0.904	**0.967**

Tables 4 and 5 illustrate the performance of the proposed system in case that the SVM output scores of individual features are fused together. The highest AUC obtained is 0.98 when the SPTS is fused with ELBP, computed on both grayscale CAPP and SAPP. When temporal filtering is not used, the highest AUC is 0.944 and achieved by the same feature combination. If we compare the highest AUC from individual (0.972) and fused features (0.98), the difference is not large. However, without temporal filtering the difference is significant (0.916 vs 0.944).

Analyzing the results of all the experiments, we can observe that in most cases, the fusion of different features (Tables 4 and 5) leads to a better result comparing to the results of individual classifiers (Tables 2 and 3). We have expected this, because when different classifiers are combined, the generalization performance of the system is improved. A single classifier may not perform well for a certain input, when it is trained with a limited dataset. For a good performance, we recommend to combine the results of multiple classifiers, instead of using single classifiers.

Furthermore, we notice that the highest scores are achieved when SPTS is combined with histogram-based features, extracted from both AAM appearance representations. This confirms that (1) geometric and appearance features have complementary information and (2) there is complementary quality in AAM representations. The use of histograms in the texture descriptors in this study results in the loss of spatial information that really depends on the person's identity. Consequently, there are subtle facial movements that cannot be captured, due to e.g. a mismatch with the applied block size. Unlike the appearance descriptors, SPTS can provide important spatial information of key facial landmarks.

Table 4. Estimated AUC (without and with SVM output averaging) for the fusion of AAM-based features with different image pre-processing methods.

Feature	Pre-processing							
	None		Grayscale		Illumination Normalization		LoG filtering	
	AUC	AUC AVG filter	AUC	AUC AVG filter	AUC	AUC AVG filter	AUC	AUC AVG filter
SPTS+SAPP	0.914	0.949	0.919	**0.956**	0.875	0.915	0.864	0.913
SPTS+CAPP	0.926	**0.959**	0.842	0.917	0.914	**0.953**	0.917	**0.955**
SAPP+CAPP	0.882	0.928	0.855	0.928	0.849	0.891	0.836	0.884
SPTS+SAPP+CAPP	0.922	0.956	0.917	0.955	0.907	0.939	0.900	0.939

Table 5. Estimated AUC (without and with SVM output averaging) for the fusion of histogram-based appearance features and AAM-based features with different image pre-processing methods.

Fused features	Pre-processing					
	Grayscale		Illumination Normalization		LoG filtering	
	AUC	AUC AVG filter	AUC	AUC AVG filter	AUC	AUC AVG filter
SPTS+LBP_SAPP+LBP_CAPP	0.942	0.970	0.937	0.970	0.926	0.962
SPTS+ELBP_SAPP+ELBP_CAPP	0.944	**0.980**	0.934	0.971	0.928	0.964
SPTS+CLBP_SAPP+CLBP_CAPP	0.942	0.974	0.935	0.968	0.924	0.963
SPTS+CELBP_SAPP+CELBP_CAPP	0.942	0.975	0.940	0.973	0.932	0.967
SPTS+LPQ_SAPP+LPQ_CAPP	0.938	0.971	0.924	0.958	0.928	0.963
SPTS+HOG_SAPP+HOG_CAPP	0.929	0.966	0.936	0.973	0.924	0.963
SPTS+ELBP_SAPP+CLBP_CAPP	0.942	0.978	0.933	0.969	0.942	**0.978**
SPTS+ELBP_SAPP+HOG_CAPP	0.930	0.969	0.935	**0.975**	0.935	0.975
SPTS+CELBP_SAPP+HOG_CAPP	0.932	0.967	0.938	**0.975**	0.937	0.970

We can conclude also, that histogram-based descriptor extraction often leads to a better discomfort detection rate, compared to the use of rough AAM appearance representations. Since the histogram-based descriptors lead to higher detection rates and additionally, their dimension is low, it is worthwhile to apply them. However, it is difficult to say which descriptor performs best, since this depends on the pre-processing method and the use of feature fusion. Furthermore, all descriptors produce quite similar results. We can certainly conclude that the combination of SPTS with histogram-based descriptors extracted from both appearance representations, leads to the best performances with high stability, irrespective of the descriptor identity. Because all histogram-based descriptors are close in performance, we prefer to choose the simplest that produces the highest rate (0.98 AUC), so that we recommend the combination of SPTS with ELBP extracted on both appearance representations. The success of (E)LBP is not surprising, since this technique is broadly accepted for face recognition applications.

With respect to the image pre-processing methods, there are no performance improvements compared to the unprocessed images. However, we should experiment with more data containing varying lighting conditions, to confirm this robustness and evaluate usefulness of the applied pre-processing. Unlike image pre-processing, temporal filtering provides always a significant improvement in the system performance (approximately 5%).

4. CONCLUSIONS

Pain and discomfort are major indications of infant illnesses. Infants are unable to indicate their pain verbally, therefore it has to be measured in another way. Currently, the medical assessment of neonatal pain considers a number of

physiological and behavioral factors, while facial expression plays a major role in pain detection. Medical staff is responsible to recognize infant pain and provide prompt relief. However, healthcare professionals are often not impartial in their judgments or fail to exploit all the available information of the neonatal expressions. With the increasing evidence of the negative effects of pain on the mental and physical health of prematurely born infants, it is of vital importance to provide an automatic monitoring system to detect pain and discomfort.

In this paper, we propose a video-based facial analysis system for discomfort detection in infants. An AAM is employed to robustly track the face in the video sequence, providing recovery in cases when the infant's face is partially or totally occluded. The presence of discomfort is detected by using AAM face representations and an SVM classifier. The detection rate of the system is improved by extracting selected histogram-based texture descriptors from the AAM appearance representations. The chosen descriptors are selected because of their low complexity and high discrimination power and are the following: Local Binary Patterns (LBP), Elongated Local Binary Patterns (ELBP), Completed Local Binary Patterns (CLBP), Completed Elongated Local Binary Patterns (CELBP), Local Phase Quantization (LPQ) and Histogram of Oriented Gradients (HOG). Further classification improvements are achieved by (1) fusing the SVM output scores of different features together and (2) by performing temporal filtering of the classification outputs. The effect of applying different image pre-processing for correcting illumination conditions and image enhancement is also examined to evaluate possible recognition improvements. Experiments are carried out with 15 videos of 8 infants. According to these experiments, the proposed system can track the face with acceptable accuracy (86.3 %) and detect discomfort with high accuracy (0.98 AUC) in the cases that the tracking is successful. The feature combination that provides the best recognition rate (0.98 AUC) is the similarity-normalized shape (SPTS) and the ELBP extracted from similarity-normalized appearance (SAPP) and canonical-normalized appearance (CAPP).

There are a number of limitations in this work that we would like to mention here. At first, the basic version of an AAM is used for the face tracking, but it works only with high accuracy when the constructed AAM is person-specific. In order to keep the tracking accuracy high and since developing a more sophisticated AAM approach requires significant effort and time, the experiments are carried out with a person-specific AAM: this means that for each infant, an AAM is constructed. However, in a real hospital setting, the overall system should be infant-independent. Secondly, this study does not take into account difficult situations where the infant's face is partly covered by plasters and breathing tubes and where the illumination conditions are poor. Thirdly, the dataset used in this study is relatively small and needs to be expanded, including also infants from different ethnicities.

In terms of future work, we suggest to focus on making the system infant-independent. We plan to develop an online AAM learning algorithm that learns the appearance of a person in the first frames of the video and then afterwards is ready to track the face in the subsequent frames. In addition to this, we aim at more extensive validation on a larger dataset prior to applying the system in a real hospital setting for clinical validation. Datasets for plastered infants and poor lighting conditions are going to be also included.

As a final conclusion, we consider our obtained results in this paper quite promising and foresee a high potential to develop an automatic discomfort detection system in real time. Since such a system can provide significant help to medical staff for infant care but also immediate relief for infants suffering from pain, this topic warrants further support for development.

REFERENCES

[1] Porter, F. L., Grunau, R. E. and Anand, K. J., "Long-term effects of pain in infants, ", J. Dev. Behav. Pediatr. 20(4), 253-261 (1999).

[2] Grunau, R. E., Holsti, L. and Peters, J. W. B., "Long-term consequences of pain in human neonates, " Semin Fetal Neonatal Med. 11(4), 268-275 (2006).

[3] Spasojević, S. and Bregun-Doronjski, A., "Pain indicators in newborns, " Med Pregl. 61(1-2), 37-42 (2008).

[4] Coffman, S., Alvarez, Y., Pyngolil, M., Petit, R., Hall, C. and Smyth, M., "Nursing assessment and management of pain in critically ill children, " Heart and Lung 26 (3) ,221-228 (1997).

[5] Ambuel, B., Hamlett, K. W., Marx, C. M. and Blumer, J. L., "Assessing distress in pediatric intensive care environments: the COMFORT scale," J. Pediatr. Psychol. 17(1), 95-109 (1992).

[6] Stevens, B., Johnston, C., Petryschen, P. and Taddio, A., "The premature infant pain profile: development and initial validation," Clinical Journal of Pain 12(1), 13-22 (1996).

[7] Holsti, L. and Grunau, R. E., "Initial validation of the Behavioral Indicators of Infant Pain (BIIP)," Pain 132(3), 264-72 (2007).
[8] Buchholz, M., Karl, H., Pomietto, M. and Lynn, A., "Pain scores in infants: a modified infant pain scale versus visual analogue," J. Pain Symptom Manage 15(2), 117-24 (1998).
[9] Brown, S., and Timmins, F., "An exploration of nurses' knowledge of, and attitudes towards, pain recognition and management in neonates," J. of Neonatal Nursing 11(2), 65-71 (2005).
[10] Brahnam, S., Chuang, C., Sexton, R. S. and Shih, F. Y., "Machine assessment of neonatal facial expressions of acute pain," Decision Support Systems 43(4), 1247-1254 (2007).
[11] Lu, G., Yuan, L., Li, X. and Li., H., "Facial Expression Recognition of Pain in Neonates," IEEE Int.Conf. on Computer Science and Software Engineering, 756-759 (2008).
[12] Han, J., Hazelhoff, L. and de With, P. H. N., "Neonatal Monitoring Based on Facial Expression Analysis," Neonatal Monitoring Technologies: Design for Integrated Solutions, IGI Global, Hersey, 303-323 (2012).
[13] Lucey, P., Cohn, J.F., Matthews, I., Lucey, S., Sridharan, S., Howlett, J. and Prkachin, K.M., "Automatically Detecting Pain in Video Through Facial Action Units," IEEE Tran. on Systems, Man, and Cybernetics, Part B: Cybernetics 41(3), 664-674 (2011).
[14] Cootes, T. F., Edwards, G. J. and Taylor, C. J., "Active appearance models," IEEE Tran. on Pattern Analysis and Machine Intelligence 23(6), 681-685 (2001).
[15] Cootes, T. F. and Taylor, C. J., "Statistical Models of Appearance for Computer Vision," Tech. Report, University of Manchester, 2000.
[16] Matthews, I. and Baker, S., "Active Appearance Models Revisited," Int. J. of Computer Vision 60(2), 135-164 (2004).
[17] Vezzaro, L., "Inverse Compositional AAM", sourceforge, 13 January 2011, http://sourceforge.net/projects/icaam/ (last access 17 November 2013).
[18] Ojala, T., Pietikainen, M. and Maenpaa, T., "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Tran. on Pattern Analysis Machine and Intelligence 24(7), 971-987 (2002).
[19] Liao, S. and Chung, A. C. S., "Face recognition by using elongated local binary patterns with average maximum distance gradient magnitude," Proc. of the 8th Asian conf. on Computer vision - Volume Part II, 672-679 (2007).
[20] Zhenhua, G., Lei, Z. and David, Z., "A completed modeling of local binary pattern operator for texture classification," IEEE Trans. on Image Proc. 19(6), 1657-1663(2010).
[21] Ojansivu, V. and Heikkila, J., "Blur insensitive texture classification using local phase quantization," Proc. of the 3rd Int. Conf. on Image and Signal Processing, 236-243 (2008).
[22] Ludwig, O., Delgado, D., Goncalves, V. and Nunes, U., "Trainable Classifier-Fusion Schemes: An Application To Pedestrian Detection," 12th Int. IEEE Conf. on Intelligent Transportation Systems, 1-6 (2009).
[23] Dalal, N. and Triggs, B., "Histograms of oriented gradients for human detection," IEEE Comp. Soc. Conf. on Computer Vision and Pattern Recognition, 886-893 (2005).
[24] Tan, X. and Triggs, B., "Enhanced Local Texture Feature Sets for Face Recognition Under Difficult Lighting Conditions," IEEE Tran. on Image Processing 19(6), 1635-1650 (2010).
[25] Cortes, C. and Vapnik, V., "Support-Vector Networks," Machine Learning 20(3), 273-297 (1995).
[26] Brummer, N. and Preez, J. d., "Application-Independent Evaluation of Speaker Detection," Computer Speech and Language 20(2-3), 230-275 (2005).
[27] Brummer, N. and de Villiers, E., "BOSARIS toolkit", December 2011, https://sites.google.com/site/bosaristoolkit/ (last access 17 November 2013).
[28] Gross, R., Matthews, I. and Baker, S., "Generic vs. person specific active appearance models," Image and Vision Computing 23(12), 1080-1093 (2005).