The implication of spatial temporal changes on facial micro-expression analysis

Walied Merghani¹ · Adrian K. Davison² · Moi Hoon Yap³

Received: 2 April 2018 / Revised: 21 December 2018 / Accepted: 26 February 2019 /
Published online: 28 March 2019
© The Author(s) 2019

Abstract

Facial micro-expression datasets lack consistency and standardisation, with different research groups using various experimental settings, in particular, where the datasets are varied in resolution and frame rates. To provide new insights into the roles of frame rate and resolution, we conduct an investigation into the use of different frame rates and resolution on current benchmark datasets (SMIC and CASME II). By using Temporal Interpolation Model, we subsample SMIC (original frame rate is 100 fps) to 50 fps and CASME II (original frame rate is 200 fps) into 100 fps and 50 fps. In addition, the resolution settings are adjusted to three scaling factors: 100% (original resolution), 75% and 50%. Three feature types are used to test the performance of these settings, which are Local Binary Patterns in Three Orthogonal Planes, 3D Histograms of Oriented Gradient and Histogram of Oriented Optical Flow. The results showed that the frame rate and resolution could affect the performance of micro-expression recognition, which behave distinctively dependent on feature types. This work provides new guidelines for future research in selecting frame rate, resolution and feature descriptors in micro-expressions recognition.

Keywords Facial micro-expression · Micro-expression recognition · Frame rate · Resolution · Temporal interpolation model

1 Introduction

There are a limited amount of datasets available for facial micro-expressions (henceforth micro-expressions) analysis, and the ones that do exist vary in standards, especially with the frame rates and resolution chosen for capturing the videos. Early datasets were created with low specification such as low resolution and frame rate. Recently with new and advanced

✉ Moi Hoon Yap
M.Yap@mmu.ac.uk

¹ Sudan University of Science and Technology, Khartoum, Sudan

² Centre for Imaging Sciences, University of Manchester, Manchester, UK

³ School of Computing, Mathematics and Digital Technology, Manchester Metropolitan University, Manchester, UK

technologies for capturing and gathering datasets, researchers start to create high quality dataset. The non-publicly available datasets include the USF-HD [23], and the Polikovsky dataset [21] which have a frame rate of 29.7 and 200 fps respectively. The publicly available datasets include the CASME [26] dataset using 60 fps, the SMIC [13] dataset using 100 fps, CASME II [27] using 200 fps, SAMM [6] using 200 fps and CAS(ME)² [22] using 30 fps. The researchers in this field have been collecting data using different settings for frame rate, resolution, experimental design, with or without stimuli, lighting condition and camera model. While some suggested high frame rate [6, 27], the most recent work [22] in this field still use a low frame rate. The question that arises here is that are these high quality datasets needed to improve micro-expressions analysis, and among those different standards which is the best?

To address the above question, we provide new insights of the implication of spatial temporal changes on micro-expression recognition by conducting a comparative study using the most popular feature types on two high frame rate and popular benchmark datasets, i.e. SMIC and CASME II. First we review the relevant spatial temporal work in micro-expression recognition outline our method in generating various frame rates and resolution. Then we summarise the three basic feature descriptors and a classifier used for this work. Finally, we present the results and discuss the future work.

2 Related work

Change in temporal settings using temporal interpolation model (TIM), which have been first introduced by Pfister et al. [18], have been tested in many micro-expression studies. To achieve more statistically stable histograms, Pfister et al. [19] used TIM to increase the number of frames of earlier version of SMIC.

In 2013, Li et al. [13] carried out a micro-expressions recognition experiment on SMIC dataset to discriminate three classes of micro-expression (positive, negative and surprise) using LBP-TOP. Due to limited number of frames in VIS and NIR, to avoid the problems that may happen when applying different parameters for LBP-TOP, TIM has been used to up-sample the frames. TIM here just used for avoiding the problem that may arise when applying LBP-TOP. A year later, Le et al. [12] used TIM to avoid the biases that can be caused by the different frame lengths. They equalized all frame lengths of CASME II and SMIC samples (15 and 10 frames respectively).

In 2015, Huang et al. [10] proposed Spatio-Temporal Local Binary Pattern with Integral Projection (STLBP-IP). They used integral projection to boost the capability of LBP-TOP with experiments conducted on the CASME II and SMIC datasets using Support Vector Machine (SVM) as a classifier. For a fair comparison with the method proposed by Li et al. [13], they used TIM to interpolate videos into 15 frames. At the same year, Li et al. [14] evaluated the performance of three feature types (LBP, HOG and histograms of image gradient orientation (HIGO)) on CASME II and SMIC. They used TIM on SMIC to test the affect of interpolation length. Different frame lengths from 10 to 80 with fixed incremental steps of 10 frames have been tested. They concluded that 10 frames achieved the best performance.

Using TIM to adjust the temporal settings is a well-known method in micro-expression analysis. Recently, there were studies investigating image resolution for deep learning by [9, 24]. However, there is a lack of thorough research in further investigating the implication of spatial-temporal changes for micro-expression recognition.

3 Methodology

Due to the inconsistency of facial micro-expression dataset specifications, such as different resolutions and frame rates, we propose to experiment the effects of these variations in micro-expression recognition. CASME II [27] and SMIC [13] has been chosen to experiment these effects because it has high frame rate which makes it intuitive to test different frame rates. It also contains a high number of micro-expression samples and higher intensity for the facial movements.

3.1 Frame rate subsampling

To subsample each micro-expression video clip to different frame rates, we use a Temporal Interpolation Model (TIM) [18]. This uses graph embedding to interpolate at random points with the micro-expression clips. This method allows for a more statistically stable feature extraction when reducing the original frame rate of SMIC and CASME II.

A micro-expression video is seen as a set of images sampled along a curve, and a continuous function is created in a low-dimensional manifold by representing the video as a path graph P_n with n vertices. The vertices correspond to video frames and edges to the adjacency matrix $\mathbf{W} \in \{0, 1\}^{n \times n}$ with $W_{i,j} = 1$ if $|i - j| = 1$ and 0 otherwise. To complete manifold embedding in the graph, P_n is mapped to a line that minimise the distance between connected vertices. If $y = (y_1, y_2, \dots, y_n)^T$ is the map, y is obtained by minimising the following

$$\sum_{i,j} (y_i - y_j)^2 W_{i,j}, \quad i, j = 1, 2, \dots, n \quad (1)$$

where this equation is equivalent to calculating the eigenvectors of the Laplacian graph P_n . The Laplacian graph is created with the eigenvectors $\{y_1, \dots, y_{n-1}\}$ and allows y_k to be viewed as a set of points described by

$$f_k^n(t) = \sin(\pi kt + \pi(n - k)/(2n)), t \in [1/n, 1] \quad (2)$$

sampled at $t = 1/n, 2/n, \dots, 1$. The resulting curve described by

$$\mathcal{F}^n(t) = \begin{bmatrix} f_1^n(t) \\ f_2^n(t) \\ \vdots \\ f_{n-1}^n(t) \end{bmatrix} \quad (3)$$

This curve is then used to temporally interpolate images at random positions within a micro-expression. To find the correspondences for the curve $\mathcal{F}^n$ within the image space, the image frames are mapped to points defined by $\mathcal{F}^n(1/n), \mathcal{F}^n(2/n), \dots, \mathcal{F}^n(1)$. A linear extension of graph embedding [25] is then used to learn a transformation vector w that minimises

$$\sum_{i,j} (w^T x_i - w^T x_j)^2 W_{i,j}, \quad i, j = 1, 2, \dots, n \quad (4)$$

where $x_i = \xi_i - \bar{\xi}$ is a mean-removed vector and ξ_i is the vectorised image. The resulting eigenvalue problem was solved by He et al. [8]

$$X L X^T w = \lambda' X X^T w \quad (5)$$

by using the singular value decomposition with $X = U \Sigma V^T$. A new image ξ can then be created using interpolation by

$$\xi = U M \mathcal{F}^n(t) + \bar{\xi} \quad (6)$$

where M is a square matrix. There is an assumption that ξ_i are linearly independent, and the validity of the TIM method depends on this.

The interpolated frames of a micro-expression clip preserves the characteristics of the original movement well, whilst smoothing out the temporal profile. For the proposed method, we chose to interpolate the original frame rate of 200 fps down to 100 and 50 fps. The amount of frames chosen was determined by

$$\alpha = \gamma(\theta \in \Omega) \quad (7)$$

where α is the amount of frames chosen for subsampling, γ is the scaling factor and $\theta \in \Omega$ is the original amount of frames θ within the movement Ω . For instance, the scaling factor for CASME II is represented by 0.5 for 100 frames and 0.25 for 50 frames.

3.2 Resolution down-scale

CASME II has been captured in 640×480 pixels in the raw section of the dataset. The pre-processed part of the dataset have about 280×340 pixels for the cropped facial area. SMIC high speed (HS) camera set to 100 fps and resolution of 640×480 was used to gather the expressions. The facial resolution for SMIC is 190×230 pixels, which has lower resolution than CASME II. In order to test the effects of resolution variations in micro-expressions recognition we scaled down both datasets (SMIC and CASME II) by 75% and 50% from the original resolution which also included as shown in Fig. 1.

3.3 Feature representation

3.3.1 Local binary pattern-three orthogonal planes(LBP-TOP)

The LBP operator forms labels for each pixel in an image by thresholding a 3×3 neighbourhood of each pixel with the centre value. The result is a binary number where if the outside pixels are equal to or greater than the centre pixel, it is assigned a 1, otherwise it is assigned a 0. The amount of labels will therefore be $2^8 = 256$ labels.

This operator was extended to use neighbourhoods of different sizes. Using a circular neighbourhood and bilinearly interpolating values at non-integer pixel coordinates allow any radius and number of pixels in the neighbourhood. The grey-scale variance of the local neighbourhood can be used as the complementary contrast method. The following notation

Fig. 1 Down scale resolution: 100% (original resolution), 75% and 50% from the original resolution

of (P, R) will be used for pixel neighbourhoods, where P are sampling points on a circle of radius R . Figure 2 shows an example of LBP computation.

Uniform patterns can be used to reduce the length of the overall feature vector and implement a single rotation-invariant descriptor. An LBP that is uniform when the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is traversed circularly. So 00000000 (0 transitions), 01110000 (2 transitions) and 11001111 (2 transitions) are uniform whereas the patterns 11001001 (4 transitions) and 01010010 (6 transitions) are not. In the computation of the LBP labels, uniform patterns are used so that there is a separate label for each uniform pattern and all the non-uniform patterns are labelled with a single label. For example, when using $(8, R)$ neighbourhood, there are a total of 256 patterns, 58 of which are uniform, which yields in 59 different labels.

Based on the LBP operator, LBP-TOP was first described as a texture descriptor [29] that used XT and YT temporal planes rather than just the 2D XY spatial plane. Yan et al. [27] used this method to report initial findings in the CASME II dataset, and Pfister et al. [19] and Davison et al. [4] used it as feature descriptors in their work.

Each region has the standard LBP operator applied [17] with c being the centre pixel and P being neighbouring pixels with a radius of R

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p \quad (8)$$

where g_c is the grey value of the centre pixel and g_p is the grey value of the p -th neighbouring pixel around R . 2^p defines weights to neighbouring pixel locations and is used to obtain the decimal value. The sign function to determine what binary value is assigned to the pattern is calculated as

$$s(A) = \begin{cases} 1, & \text{if } A \geq 0 \\ 0, & \text{if } A < 0 \end{cases} \quad (9)$$

If the grey value of P is larger than or equal to c , then the binary value is 1, otherwise it will be 0. Figure 2 illustrates the sign function on a neighbourhood of pixels. After the image has been assigned LBP, the histogram can be calculated by

$$H_i = \sum_{x,y} I\{LBP_l(x, y) = i\}, i = 0, \dots, n - 1 \quad (10)$$

where $LBP_l(x, y)$ is the image labelled with LBP. As this method is incorporating temporal data, the histogram can be extended to be calculated for all three planes

$$H_{i,j} = \sum_{x,y,t} I\{LBP_j(x, y, t) = i\}, i = 0, \dots, n_j - 1 \quad (11)$$

Fig. 2 LBP code calculation by using the difference of the neighbourhood pixels around the centre

where n_j is the number of labels produced by the LBP operator in the j th plane. $j = 0, 1, 2$ which represents the XY, XT and YT planes respectively. $LBP_i(x, y, t)$ expresses the LBP code of the central pixel (x, y, t) in the j th plane. The $I\{\mathbf{A}\}$ function is the equivalent to (10) that refers to the sign function in (9). An illustration of the LBP-TOP histogram concatenation process can be seen in Fig. 3.

The neighbouring points and radius parameters (P, R) can be defined as $P_{XY}, P_{XT}, P_{YT}, R_X, R_Y, R_T$ for each plane and axis, with the overall feature descriptor defined as $LBPTOP_{P_{XY}, P_{XT}, P_{YT}, R_X, R_Y, R_T}$. We chose to use the best case results from [27] and set the neighbouring points and radii parameters to $LBPTOP_{4,4,4,1,1,4}$.

3.3.2 3D histograms of oriented gradient (3DHOG)

3D histograms of oriented gradient (3DHOG) [11] is adapted version of histograms of oriented gradient (HOG) for static images to be suitable for dynamic texture in micro-expressions. 2D can be represent as $I(x, y)$, orientation and gradient could be calculated [5, 21] as follows

$$\begin{aligned} m_{2D}(x, y) &= \sqrt{\delta I_x(x, y)^2 + \delta I_y(x, y)^2} \\ \theta_{2D}(x, y) &= \tan^{-1}(\delta I_y(x, y)^2 / \delta I_x(x, y)^2) \end{aligned} \quad (12)$$

where $\delta I_x(x, y)$ and $\delta I_y(x, y)$ stand for image partial derivative. In 3D case the video $v(x, y, t)$ where t refer to time, firstly partial derivative should be calculated along x, y and t . then compute magnitude $m_{xy}(x, y, t)$, $m_{xt}(x, y, t)$ and $m_{yt}(x, y, t)$ and orientation $\theta_{xy}(x, y, t)$, $\theta_{xt}(x, y, t)$ and $\theta_{yt}(x, y, t)$ for each couple $(\delta v_x, \delta v_y)$, $(\delta v_x, \delta v_t)$ and $(\delta v_y, \delta v_t)$ using (13) [21]

$$\begin{aligned} m_{xy}(x, y, t) &= \sqrt{\delta v_x(x, y, t)^2 + \delta v_y(x, y, t)^2} \\ \theta_{xy}(x, y, t) &= \tan^{-1} \left(\frac{\delta v_x(x, y, t)^2}{\delta v_y(x, y, t)^2} \right) \\ m_{yt}(x, y, t) &= \sqrt{\delta v_y(x, y, t)^2 + \delta v_t(x, y, t)^2} \\ \theta_{yt}(x, y, t) &= \tan^{-1} \left(\frac{\delta v_y(x, y, t)^2}{\delta v_t(x, y, t)^2} \right) \\ m_{xt}(x, y, t) &= \sqrt{\delta v_x(x, y, t)^2 + \delta v_t(x, y, t)^2} \\ \theta_{xt}(x, y, t) &= \tan^{-1} \left(\frac{\delta v_x(x, y, t)^2}{\delta v_t(x, y, t)^2} \right) \end{aligned} \quad (13)$$

Fig. 3 LBP is calculated on every block in all three planes. Each plane is then concatenated to obtain the final LBP-TOP feature histogram

gradient orientation histograms are computed for every frame for the couples $(\delta v_x, \delta v_y)$ gradient orientation histogram contains 8 bins $(\delta v_x, \delta v_t)$ and $(\delta v_y, \delta v_t)$ contains 12 bins. After computing histograms in $(\delta v_x, \delta v_y)$ $(\delta v_x, \delta v_t)$ and $(\delta v_y, \delta v_t)$ for every frame, all histograms corresponding to the same frame are concatenated to one feature vector and normalized.

3.3.3 Histogram of oriented optical flow (HOOF)

HOOF [2] features compute optical flow for each frame, then the vector binned according to the orientation and weighted according to the magnitude, where each optical flow consist of pair of angle and magnitude and can be represented as $v = [x, y]^T$ with direction $\theta = \tan^{-1}(\frac{y}{x})$. Figure 4 explain how to build HOOF feature with 4 bins.

One of the HOOF-based methods is Main Directional Mean Optical Flow (MDMO) [15] which is a ROI-based normalised statistic feature. Discriminative Response Map Fitting (DRMF) [1] used to locate 68 facial feature points, 66 of them have been used (two inner corner points of lip ignored) to normalize faces base on the first frame. Normalized face has been partitioned to 36 regions of interest (ROIs) determined by 66 feature points and partially based on FACS. Using optical flow the change in intensity between two pixels has been detected between two frames over time motion of objects. Changing in intensity can be represented by

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \quad (14)$$

the optical flow value of a pixel between two frames at time t in Euclidean coordinates can be represented as a two-dimensional vector

$$[V_x^t, V_y^t]^T \quad (15)$$

To compute MDMO feature for micro-expression recognition the Euclidean coordinates $[V_x^t, V_y^t]^T$ has been converted into polar coordinates (ρ_i, θ_i) , where ρ_i is the magnitude and θ_i is orientation of the optical flow vectors. Histogram of oriented optical flow (HOOF) [2] has been computed for each ROI in each frame R_i^k , where i the index of frames and k is the index of ROIs and the optical has been classified into 8 bins. A mean vector $\bar{u}_i^k$ has been computed for optical flow vectors in bin with maximum count. $\bar{u}_i^k = (\bar{\rho}_i^k, \bar{\theta}_i^k)$, $\bar{\theta}_i^k$ called *main direction*. A feature vector Ψ_i has been built by $\Psi_i = (\bar{u}_i^1, \bar{u}_i^2, \dots, \bar{u}_i^{36})$, this make The dimension of feature vector is $36 \times 2 = 72$, where 36 is the number of ROIs. Then micro-expression represented by concatenated features vector of each frame $\Gamma = (\Psi_1, \Psi_2, \dots, \Psi_n)$,

Fig. 4 Build HOOF features with 4 bins

where n the number of frames in micro-expression. Finally, a normalisation in Cartesian coordinate has been done before converted back to polar coordinates and represented by

$$\overline{\Psi} = [(\overline{\rho_1}, \overline{\theta_1})^T, (\overline{\rho_2}, \overline{\theta_2})^T, \dots, (\overline{\rho_{36}}, \overline{\theta_{36}})^T] \quad (16)$$

In this experiment, block-based HOOF has been used and the parameters was set to $pRow = 6$, $pCol = 6$ and $pFrames = 6$ where $pRow$, $pCol$ and $pFrames$ is sub-block size in pixels for the row, column and frame respectively. The number of blocks was set to 3×3 spatial blocks and 2 temporal blocks, Horn-Schunck method has been used for optical flow computing and the last parameter is quantization, which is set to 8 orientations.

3.4 Classification

First proposed by Cortes and Vapnik [3], a Support Vector Machine (SVM) attempts to find a linear decision surface (hyperplane) that can separate classes and has the largest distance between support vectors (elements in data closest to each other across classes). If a linear surface does not exist, then an SVM is able to use kernel functions to map the data into a higher dimensional space where a decision surface can be found.

We use the Sequential Minimal Optimization (SMO) [20] algorithm to train the SVMs. SMO is able to break down large quadratic programming problems into a series of the smallest possible problems, which are solved analytically and avoids using a time-consuming numerical quadratic programming optimisation as an inner loop. SMO is also able to handle large training sets and is one of the computationally fastest methods of evaluating linear SVMs.

3.5 Evaluation method

We report the result in F-Measure (or F1-Score) due to the imbalanced database [12]. Using the conventional accuracy measure may result in a bias towards classes with large number of samples, hence overestimating the capability of the method. F-Measure micro-average across the whole dataset and is computed based on the total true positives, false negatives and false positives, across 10-fold cross validation and/or Leave-one-subject-out (LOSO) folds. The next section discuss and compare the result of our comparative study (Table 1).

4 Results and discussion

We have conducted comprehensive evaluation on the performance of three popular feature representations on micro-expressions recognition using difference frame rates and resolution. We observed that the top performers across different categories were varied.

Table 2 summarised the results of two validation methods, 10-fold cross validation and Leave-one-subject-out (LOSO), on CASME II. For 10-fold cross validation, the best result was LBP-TOP with 200 fps and 100% of the original resolution, achieved an F-Measure of 0.637. However, when validated with LOSO, the best results was HOOF with 200 fps and 50% resolution, achieved an F-Measure of 0.439.

Table 3 summarised the results of 10-fold cross validation and LOSO on SMIC. For 10-fold cross validation, the best result was 3DHOG with 50 fps and 75% resolution, achieved an F-Measure of 0.624. When validated with LOSO, the best result was HOOF with 100 fps and 75% resolution, achieved an F-Measure of 0.614.

Table 1 The performance of the state-of-the-art methods on SMIC and CASME II

Method	SMIC	CASME II
Li et al. [14]	0.5352	0.5749
Lu et al. [16]	0.8286	0.6419
Le et al. [12]	0.4434	0.4378
Huang et al. [10]	0.5793	0.5951
Liu et al. [15]	0.8000	0.6737
LBP-TOP	0.561	0.796
3DHOG	0.538	0.611
HOOF	0.593	0.636

Note that we have only included some popular previous works that reported their results on both datasets

4.1 Comparison of the state-of-the-art methods

Table 1 compared the performance of the state-of-the-art methods on SMIC and CASME II. The majority of the state-of-the-art results were based on the original resolution and frame-rate. Therefore, this comparison was based on the original properties of the dataset. In addition, the majority of previous works reported their results using accuracy as the performance metric. Therefore, we compared these methods based on accuracy. From our observation, although Li et al. [14] and Liu et al. [15] achieved good accuracy on SMIC dataset, the performance on CASME II were comparable to the basic features of HOOF, LBP-TOP and 3DHOG.

4.2 Temporal analysis

Since LOSO is a better approach in performance measure, we further analyse its performance for temporal analysis. As shown in Figs. 5 and 6, we observed that LBP-TOP and HOOF performed better in high frame rate on CASME II and SMIC. For the majority of the resolution, F-Measure decreased or maintained as the frame rates dropped. In contrast, when compared the performance of 3DHOG on different frame rates, the F-Measure increased on lower frame rate (50 fps for CASME II and SMIC), as illustrated in Fig. 7.

4.3 Spatial analysis

Regarding the effect on varying resolutions on micro-expressions recognition, there is also an inconsistency for optimal resolution through different features. We found that LBP-TOP achieved better result when the full resolution of CASME II and SMIC were used. For HOOF, a low resolution (50% from the original resolution) achieved the best results on CASME II and the mid resolution (75% from the original resolution) achieved the best results on SMIC. This might be due to the lower facial resolution of SMIC when compared to CASME II. For 3DHOG, the result on spatial analysis is inconclusive. As illustrated in Fig. 7, the results are varied but the majority of the results performed better with low frame rate.

4.4 Features analysis

By taking into account the effect of resolution and frame rate on micro-expressions recognition when using different feature descriptors, we observed that LBP-TOP as an example

Table 2 The results of CASME II for both the 10-fold cross-validation and leave-one-subject-out for the 3DHOG, HOOF and LBP-TOP features with a varying resolutions and frame rates. The best result for each metric is in bold

		10-Fold Cross-Validation				Leave-One-Subject-Out (LOSO)			
Resolution	Frame rate	Accuracy	TPR	FPR	F-Measure	Accuracy	TPR	FPR	F-Measure
3DHOG									
100%	200	0.718	0.478	0.290	0.425	0.611	0.368	0.271	0.319
	100	0.713	0.474	0.295	0.421	0.595	0.345	0.276	0.301
	50	0.731	0.509	0.281	0.463	0.627	0.396	0.266	0.348
	200	0.722	0.486	0.286	0.436	0.595	0.341	0.265	0.291
75%	100	0.714	0.474	0.293	0.423	0.590	0.345	0.270	0.295
	50	0.727	0.501	0.284	0.456	0.604	0.356	0.258	0.301
	200	0.727	0.494	0.279	0.446	0.594	0.352	0.267	0.308
50%	100	0.723	0.490	0.283	0.444	0.581	0.345	0.291	0.292
	50	0.735	0.509	0.269	0.472	0.603	0.368	0.272	0.311
HOOF									
100%	200	0.698	0.475	0.319	0.423	0.636	0.427	0.304	0.383
	100	0.679	0.443	0.374	0.318	0.615	0.404	0.345	0.314
	50	0.676	0.443	0.365	0.362	0.616	0.404	0.346	0.317
	200	0.708	0.494	0.305	0.450	0.627	0.408	0.275	0.368
75%	100	0.666	0.420	0.389	0.285	0.614	0.392	0.329	0.309
	50	0.673	0.435	0.357	0.367	0.609	0.376	0.315	0.307
	200	0.744	0.553	0.258	0.536	0.668	0.475	0.283	0.439
50%	100	0.720	0.518	0.318	0.455	0.610	0.388	0.326	0.321
	50	0.673	0.431	0.357	0.357	0.580	0.365	0.332	0.297
LBP-TOP									
100%	200	0.819	0.635	0.131	0.637	0.660	0.435	0.193	0.415
	100	0.802	0.603	0.139	0.609	0.610	0.384	0.238	0.348
	50	0.779	0.552	0.166	0.553	0.618	0.360	0.222	0.346
	200	0.801	0.600	0.144	0.603	0.613	0.376	0.240	0.354
75%	100	0.802	0.603	0.146	0.606	0.595	0.352	0.256	0.318
	50	0.772	0.545	0.172	0.545	0.620	0.368	0.220	0.342
	200	0.778	0.568	0.168	0.571	0.618	0.380	0.202	0.346
50%	100	0.781	0.572	0.163	0.575	0.600	0.348	0.210	0.322
	50	0.763	0.533	0.186	0.533	0.598	0.325	0.219	0.297

of texture-based features performed better on high resolution and high frame rate as illustrated in Fig. 5, texture-based features depended on pixels to extract the informations of micro-expressions, so more pixels(high resolution) means more informations. Also, high frame rate increases the number of pixels on the 3rd plane, this is why LBP-TOP performs better on high settings. Gradient-based features such as 3DHOG in these experiments does

Table 3 The results of SMIC for both the 10-fold cross-validation and leave-one-subject-out for the 3DHOG, HOOF and LBP-TOP features with varying resolutions and frame rates. The best result for each metric is in bold

		10-Fold Cross-Validation				Leave-One-Subject-Out (LOSO)			
Resolution	Frame rate	Accuracy	TPR	FPR	F-Measure	Accuracy	TPR	FPR	F-Measure
3DHOG									
100%	HS(100 FPS)	0.720	0.597	0.250	0.588	0.538	0.378	0.274	0.333
	HS(50 FPS)	0.716	0.591	0.245	0.586	0.567	0.426	0.320	0.407
	HS(100 FPS)	0.707	0.579	0.261	0.569	0.594	0.469	0.265	0.429
75%	HS(50 FPS)	0.739	0.628	0.230	0.624	0.533	0.402	0.313	0.374
	HS(100 FPS)	0.721	0.591	0.242	0.580	0.556	0.408	0.252	0.405
50%	HS(50 FPS)	0.723	0.597	0.233	0.594	0.593	0.457	0.264	0.458
HOOF									
100%	HS(100 FPS)	0.734	0.621	0.231	0.619	0.593	0.493	0.278	0.498
	HS(50 FPS)	0.713	0.597	0.264	0.586	0.574	0.475	0.357	0.467
	HS(100 FPS)	0.722	0.603	0.243	0.599	0.687	0.609	0.254	0.614
75%	HS(50 FPS)	0.698	0.573	0.275	0.564	0.619	0.536	0.372	0.523
	HS(100 FPS)	0.723	0.609	0.259	0.598	0.627	0.530	0.310	0.534
50%	HS(50 FPS)	0.683	0.554	0.305	0.521	0.609	0.506	0.376	0.486
LBP-TOP									
100%	HS(100 FPS)	0.703	0.573	0.241	0.572	0.561	0.402	0.279	0.349
	HS(50 FPS)	0.695	0.560	0.241	0.561	0.567	0.402	0.294	0.349
	HS(100 FPS)	0.693	0.554	0.244	0.554	0.549	0.384	0.285	0.344
75%	HS(50 FPS)	0.673	0.524	0.267	0.522	0.538	0.384	0.345	0.335
	HS(100 FPS)	0.700	0.567	0.247	0.566	0.545	0.402	0.268	0.395
50%	HS(50 FPS)	0.701	0.567	0.245	0.564	0.488	0.317	0.306	0.313

Fig. 5 Comparison of F-Measure using LBP-TOP with varying resolution and frame-rates. The graph shows the best result is using high frame rate and high resolution when evaluated using LOSO validation on CASME II (100%, 75% and 50%) and SMIC (HS100%, HS75% and HS50%)

Fig. 6 Comparison of F-Measure using HOOF with varying resolution and frame-rates. The graph shows the best result is using high frame rate and lower resolution when evaluated using LOSO validation on CASME II (100%, 75% and 50%) and SMIC (HS100%, HS75% and HS50%)

not need a high specification to achieve good results as shown in Fig. 7, with the best result achieved on 50 fps across different resolutions. On the other hand, HOOF features, which is optical flow-based, as shown in Fig. 6, performed better in a high frame rate scenario. Whilst there was no considerable need for high resolution, we found that the best result was recorded in 50% of the original resolution. As the opposite of texture-based features, we found that optical flow extracts the information by calculating the motion between frames, so it depends on temporal more than spatial this is why it needs high frame rate for better performance rather than high resolution.

4.5 Summary

The LOSO method of evaluation shows a decrease in accuracy compared with the 10-fold cross validation. While both have been used in previous research, the differences show the

Fig. 7 Comparison of F-Measure using 3DHOG with varying resolution and frame-rates. The graph shows the overall best result is achieved by using low frame rate when evaluated using LOSO validation on CASME II (100%, 75% and 50%) and SMIC (HS100%, HS75% and HS50%)

challenging nature of finding the correct way to determine a method's success. Further, the lower performance seen overall with LOSO can be attributed to the fairer nature of testing on a subject completed omitted from the training stage.

The frame rate is certainly important, with drops seen as this is decreased. However, obtaining equipment and data storage for 200 FPS recording can be difficult. A good trade-off could reduce the frame rate, but keep the best performing resolution and feature.

LBP-TOP and 3DHOG are relatively simple feature types, with HOOF being somewhat more informative based on temporal data. As micro-expression movements can look unique, even though the same muscle are used, simple features tend to pick out the obvious changes and struggle to model how a real micro-expression differs from noise.

5 Conclusion

Research on automated facial micro-expression recognition using machine learning has witnessed good progress in recent years. A number of promising methods based on texture features, gradient features and optical flow features have been proposed. Many datasets were generated but lack of standardisation is indeed a great challenge. Therefore, this paper focused on comparing and discussing the effect of different frame rate and resolution.

Three of the most famous feature descriptors have been used to represent micro-expression. These features have variation in their nature, which is very suitable to test the effect of frame rate and resolution based on these features. LBP-TOP used as an examples of texture-based features, 3DHOG to represent gradient-based features and HOOF to represent optical flow-based features. For LBP-TOP we found it was perform better in high-resolution and high frame rate, unlike 3DHOG which perform better in low specification, high frame rate improved recognition rate when HOOF features used but increasing resolution does not give a significant improvement. Based on the results obtained, we found that the results of micro-expressions recognition with different settings for datasets being influenced by different features used, where the different features act in a different way through various settings. Analysis on 10 folds of 10-fold cross validation have been done, the distribution of classes through 10 folds are well distributed with low standard deviation, this implies that the sampling variations are minimal as shown in the [Appendix](#). Based on this analysis, the results variation were caused by the spatial and temporal changes.

The accuracy of many state-of-the-art methods is still too low to be deployed effectively in a real-world environment. We provide important insights for researchers in this field to consider the settings when conducting new experiment in the future. The progress in research of micro-expressions recognition can aid in the paradigm shift in affect computing for real-world applications in psychology, health study and security control. Future work will be investigating into SAMM dataset, where the authors [7] recently introduced emotional classes and objective classes. This dataset has been used in micro-expressions grand challenge [28].

Open Access This article is distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Appendix: Sampling variation

Table 4 The sampling variation of classes distribution through the 10 folds

Classes	1	2	3	4	5
Fold 1	2	3	4	5	11
Fold 2	1	7	5	0	12
Fold 3	3	6	2	2	12
Fold 4	3	1	0	4	17
Fold 5	3	8	4	0	10
Fold 6	12	5	2	2	4
Fold 7	2	10	0	8	5
Fold 8	3	5	5	0	12
Fold 9	1	6	0	6	12
Fold 10	2	8	3	0	12
STD	3.19	2.60	2.01	2.90	3.74

References

1. Asthana A, Zafeiriou S, Cheng S, Pantic M (2013) Robust discriminative response map fitting with constrained local models. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3444–3451
2. Chaudhry R, Ravichandran A, Hager G, Vidal R (2009) Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In: IEEE conference on computer vision and pattern recognition, 2009. CVPR 2009. IEEE, pp 1932–1939
3. Cortes C, Vapnik V (1995) Support-vector networks. *Mach Learn* 20(3):273–297
4. Davison AK, Yap MH, Costen N, Tan K, Lansley C, Leightley D (2014) Micro-facial movements: an investigation on spatio-temporal descriptors. In: Computer vision-ECCV 2014 workshops. Springer, pp 111–123
5. Davison A, Merghani W, Lansley C, Ng CC, Yap MH (2018) Objective micro-facial movement detection using face-based regions and baseline evaluation. In: 2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018). IEEE, pp 642–649
6. Davison AK, Lansley C, Costen N, Tan K, Yap MH (2018) Samm: a spontaneous micro-facial movement dataset. *IEEE Trans Affect Comput* 9(1):116–129. <https://doi.org/10.1109/TAFFC.2016.2573832>
7. Davison A, Merghani W, Yap M (2018) Objective classes for micro-facial expression recognition. *J Imaging* 4(10):119. Multidisciplinary Digital Publishing Institute
8. He X, Cai D, Yan S, Zhang HJ (2005) Neighborhood preserving embedding. In: Tenth IEEE international conference on computer vision, 2005. ICCV 2005, vol 2. IEEE, pp 1208–1213
9. He K, Zhang X, Ren S, Sun J (2014) Spatial pyramid pooling in deep convolutional networks for visual recognition. In: European conference on computer vision. Springer, pp 346–361
10. Huang X, Wang SJ, Zhao G, Pietikainen M (2015) Facial micro-expression recognition using spatiotemporal local binary pattern with integral projection. In: Proceedings of the IEEE international conference on computer vision workshops, pp 1–9
11. Klaser A, Marszałek M, Schmid C (2008) A spatio-temporal descriptor based on 3d-gradients. In: BMVC 2008-19th British machine vision conference. British Machine Vision Association, pp 275–1
12. Le Ngo AC, Phan RCW, See J (2014) Spontaneous subtle expression recognition: imbalanced databases and solutions. In: Computer vision-ACCV 2014. Springer, pp 33–48
13. Li X, Pfister T, Huang X, Zhao G, Pietikainen M (2013) A spontaneous micro-expression database: inducement, collection and baseline. In: 2013 10th IEEE international conference and workshops on automatic face and gesture recognition (FG). IEEE, pp 1–6
14. Li X, Hong X, Moilanen A, Huang X, Pfister T, Zhao G, Pietikainen M (2015) Reading hidden emotions: spontaneous micro-expression spotting and recognition. [arXiv:1511.00423](https://arxiv.org/abs/1511.00423)
15. Liu YJ, Zhang JK, Yan WJ, Wang SJ, Zhao G, Fu X (2015) A main directional mean optical flow feature for spontaneous micro-expression recognition. *IEEE Transaction of Affective Computing*

16. Lu Z, Luo Z, Zheng H, Chen J, Li W (2014) A delaunay-based temporal coding model for micro-expression recognition. In: Asian Conference on computer vision. Springer, pp 698–711
17. Ojala T, Pietikäinen M, Mäenpää T (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans Pattern Anal Mach Intell* 24(7):971–987. <https://doi.org/10.1109/TPAMI.2002.1017623>
18. Pfister T, Li X, Zhao G, Pietikäinen M (2011) Differentiating spontaneous from posed facial expressions within a generic facial expression recognition framework. In: 2011 IEEE international conference on computer vision workshops (ICCV workshops). IEEE, pp 868–875
19. Pfister T, Li X, Zhao G, Pietikäinen M (2011) Recognising spontaneous facial micro-expressions. In: 2011 IEEE international conference on computer vision (ICCV). IEEE, pp 1449–1456
20. Platt J et al (1999) Fast training of support vector machines using sequential minimal optimization
21. Polikovsky S, Kameda Y, Ohta Y (2009) Facial micro-expressions recognition using high speed camera and 3d-gradient descriptor. In: 3rd International conference on crime detection and prevention (ICDP 2009). IET, pp 1–6
22. Qu F, Wang SJ, Yan WJ, Li H, Wu S, Fu X (2017) Cas (me)²: a database for spontaneous macro-expression and micro-expression spotting and recognition. *IEEE Transactions on Affective Computing*
23. Shreve M, Godavarthy S, Goldgof D, Sarkar S (2011) Macro- and micro-expression spotting in long videos using spatio-temporal strain. In: 2011 IEEE international conference on automatic face gesture recognition and workshops (FG 2011), pp 51–56. <https://doi.org/10.1109/FG.2011.5771451>
24. Wu L, Wang Y, Li X, Gao J (2018) Deep attention-based spatially recursive networks for fine-grained visual recognition. *IEEE Transactions on Cybernetics*
25. Yan S, Xu D, Zhang B, Zhang HJ, Yang Q, Lin S (2007) Graph embedding and extensions: a general framework for dimensionality reduction. *IEEE Trans Pattern Anal Mach Intell* 29(1):40–51
26. Yan WJ, Wu Q, Liu YJ, Wang SJ, Fu X (2013) Casme database: a dataset of spontaneous micro-expressions collected from neutralized faces. In: 2013 10th IEEE International conference and workshops on automatic face and gesture recognition (FG). IEEE, pp 1–7
27. Yan WJ, Li X, Wang SJ, Zhao G, Liu YJ, Chen YH, Fu X (2014) Casme ii: an improved spontaneous micro-expression database and the baseline evaluation. *PloS one*, 9(1)
28. Yap MH, See J, Hong X, Wang S-J (2018) Facial micro-expressions grand challenge 2018 summary. In: 2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018). IEEE, pp 675–678
29. Zhao G, Pietikainen M (2007) Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Trans Pattern Anal Mach Intell* 29(6):915–928. <https://doi.org/10.1109/TPAMI.2007.1110>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Walied Merghani received his BSc (Hons.) degree in Computer and Information System from Sudan University of Science and Technology (SUST), Sudan in 2006. He received the MSc degree in Computer Science from SUST in 2013. He is now a PhD student at SUST. He is a Lecturer in Computer Science department at SUST. His research interests include microfacial expression recognition, pattern recognition and handwriting recognition.

Adrian K. Davison is a Research Associate at the University of Manchester. He received his BSc (Hons.) degree in Multimedia Computing in 2012 and his PhD degree in 2016 from Manchester Metropolitan University. He had an active role as a student representative within MMU, Co-Chairing the internal MMU Science and Engineering Symposium 2015. His current role is the medical imaging analysis of children's hip disease. His research expertise is in facial expression and micro-facial expression analysis, deep learning and medical image analysis.

Moi Hoon Yap is Reader in Computer Science at the Manchester Metropolitan University and a Royal Society Industry Fellow with Image Metrics Ltd. She received her PhD in Computer Science from Loughborough University in 2009. After her PhD, she worked as Postdoctoral Research Assistant (April 09 - Oct 11) in the Centre for Visual Computing at the University of Bradford. She serves as an Associate Editor for Journal of Open Research Software and reviewers for IEEE transactions/journals (Image Processing, Multimedia, Cybernetics, biomedical health and informatics). Her research expertise is in facial analysis and medical image analysis.