Edge-based Foreground Detection with Higher Order Derivative Local Binary Patterns for Low-resolution Video Processing

Francis Deboeverie, Gianni Allebosch, Dirk Van Haerenborgh, Peter Veelaert and Wilfried Philips
Department of Telecommunications and Information Processing, Image Processing and Interpretation, UGent/iMinds, St-Pietersnieuwstraat 41, 9000 Ghent, Belgium.

Keywords: Foreground Segmentation, Edge Detection, Local Binary Patterns, Low-resolution Video Processing.

Abstract: Foreground segmentation is an important task in many computer vision applications and a commonly used approach to separate foreground objects from the background. Extremely low-resolution foreground segmentation, e.g. on video with resolution of 30x30 pixels, requires modifications of traditional high-resolution methods. In this paper, we adapt a texture-based foreground segmentation algorithm based on Local Binary Patterns (LBPs) into an edge-based method for low-resolution video processing. The edge information in the background model is introduced by a novel LBP strategy with higher order derivatives. Therefore, we propose two new LBP operators. Similar to the gradient operator and the Laplacian operator, the edge information is obtained by the magnitudes of First Order Derivative LBPs (FOD-LBPs) and the signs of Second Order Derivative LBPs (SOD-LBPs). Posterior to background subtraction, foreground corresponds to edges on moving objects. The method is implemented and tested on low-resolution images produced by monochromatic smart sensors. In the presence of illumination changes, the edge-based method outperforms texture-based foreground segmentation at low resolutions. In this work, we demonstrate that edge information becomes more relevant than texture information when the image resolution scales down.

1 INTRODUCTION

Foreground/background segmentation is an essential pre-processing step, aimed at the separation of moving objects, the foreground (FG), from an expected scene, the background (BG). FG/BG segmentation is often considered in a broader context of, for instance, human activity detection (Grünwedel et al., 2013). The focus of FG/BG segmentation has drifted towards higher resolutions, yielding new, high-performing, algorithms (Zivkovic, 2004; Heikkilä and Pietikäinen, 2006; Barnich and Droogenbroeck, 2009; Grüenwedel et al., 2011). However, the price of processing on high-resolution smart cameras has increased (Camilli and Kleihorst, 2011), which makes multicamera systems prohibitive for a low price range. On the contrary, low-resolution image sensors allow very low-cost processing and can therefore allow more sensors at a lower total system cost (Camilli and Kleihorst, 2011). For instance, low-resolution imagers have been proven useful in (Hengstler and Aghajan, 2006). However, existing FG/BG segmentation algorithms need to be adapted for optimal operation on extremely low-resolution video (e.g. 30x30 pixels), since they can only scale down to resolutions of about 128x128 pixels (Grünwedel et al., 2011).

An approach that is promising to work at low resolutions uses Local Binary Patterns (LBPs) (Ojala et al., 1996; Ojala et al., 2002). The LBP operator describes each pixel by the relative grey levels of its neighbouring pixels. The binary patterns or their statistics, most commonly the histogram, are then used for further image analysis. In this sense, the BG subtraction technique in (Heikkilä and Pietikäinen, 2006) is a region-based method describing local texture characteristics as a modification of the LBPs (Ojala et al., 1996). Each pixel is modelled as a group of adaptive local binary pattern histograms that are calculated over a circular region around the pixel.

In this paper, we adapt the texture-based FG segmentation algorithm based on LBPs (Heikkilä and Pietikäinen, 2006) into an edge-based method for low-resolution video processing. In the proposed BG subtraction method, FG corresponds to edges on moving objects. This idea is based on the concept of detecting FG in video by moving intensity differences in the human brain (Movshon et al., 1986). The edge information in the BG model is introduced by a novel generic strategy with higher order derivative LBPs. Therefore, we introduce two new LBP operators. Re-

Deboeverie F., Allebosch G., Van Haerenborgh D., Veelaert P. and Philips W..
Edge-based Foreground Detection with Higher Order Derivative Local Binary Patterns for Low-resolution Video Processing.
DOI: 10.5220/0004723403390346
In *Proceedings of the 9th International Conference on Computer Vision Theory and Applications* (VISAPP-2014), pages 339-346
ISBN: 978-989-758-003-1

lated to the gradient edge operator and the Laplacian edge operator, the key idea is to build a BG model from the magnitudes of First Order Derivative LBPs (FOD-LBPs) and the signs of Second Order Derivative LBPs (SOD-LBPs). Edge detection uses the magnitude of the first derivative to detect the presence of an edge at a point in an image (i.e., to determine if a point is on a ramp). Similarly, the sign of the second derivative is used to determine whether a pixel lies on a dark or light side of an edge.

In this work, we first evaluate the FOD-LBP operator and the SOD-LBP operator for edge detection. Next, we compare the results of the proposed FG/BG segmentation method with the results of the state-of-the-art technique in (Heikkilä and Pietikäinen, 2006). The method is implemented and tested on low-resolution images produced by monochromatic smart sensors. In the experiments, we consider varying illumination conditions and video resolutions. When evaluated against ground truth, we will show that the edge-based method outperforms texture-based FG segmentation at low resolutions. An important contribution in this work consists of the demonstration that edge information is more important than texture information at low resolutions. We will show that lowering the image resolution quadratically reduces texture information and linearly reduces edge information.

1.1 Related Work

The LBP operator has been succesfully applied to various computer vision problems such as face recognition (Ahonen et al., 2006), BG subtraction (Heikkilä and Pietikäinen, 2006), recognition of 3D textured surfaces (Pietikäinen et al., 2004) and describing interest regions (Heikkilä et al., 2009). The LBP has properties that favor its usage in this work, such as tolerance against illumination changes and computational simplicity.

In some studies edge detection has been used prior to LBP computation to enhance the gradient information. Yao and Chen (Yao and Chen, 2003) proposed local edge patterns (LEP) to be used with color features for color texture retrieval. In LEP, the Sobel edge detection and thresholding are used to find strong edges, and then LBP-like computation is used to derive the LEP patterns. In their method for shape localization Huang el al. (Huang et al., 2004) proposed an approach in which gradient magnitude images and original images are used to describe the local appearance pattern of each facial keypoint. A derivative-based LBP is used by applying LBP computation to the gradient magnitude image obtained by a Sobel operator. The Sobel-LBP later proposed by Zhao et al. (Zhao et al., 2008) uses the same idea for facial image representation. First the Sobel edge detector is used and the LBPs are computed from the gradient magnitude images. The difference with our work is that above-mentioned methods compute traditional LBPs of edge images, while our method gathers the edge information from novel LBP operators.

Inspired by LBP, higher order local derivative patterns (LDP) were proposed by Zhang et al., with applications in face recognition (Zhang et al., 2010). The patterns extracted by LDP will provide more detailed information, but may also be more sensitive to noise than in LBP. In above-mentioned work the focus is on texture, while in this work the focus is on edges.

Besides the FG segmentation algorithm based on LBPs (Heikkilä and Pietikäinen, 2006), there exist several other techniques for FG/BG segmentation. We give an overview of the most recent and best performing algorithms. The Gaussian Mixture Model (GMM) method of (Zivkovic, 2004) uses a variable number of Gaussians to model the color value distribution of each pixel as a multi-modal signal. This parametric approach adapts the model parameters to statistical changes. ViBe (Barnich and Droogenbroeck, 2009) is a sample-based approach for modeling the color value distribution of pixels. The sample set is updated according to a random process that substitutes old pixel values for new ones. In (Grüenwedel et al., 2011), the FG is subtracted from the BG by detecting moving Sobel edges. Edge dependencies are used as statistical features of FG and BG regions and FG is defined as regions containing moving edges, and BG as regions containing static edges in the scene. The BG modeling uses gradient estimates in *x* and *y*-direction. Above-mentioned algorithms have limited opportunities to scale down to low resolutions.

The remainder of this paper is organized as follows: Section 2 describes a novel strategy to extract edge-based LBPs. Firstly, subsection 2.1 summarizes the basic concepts of LBPs. Secondly, subsection 2.2 presents higher order derivative LBP edge operators, more specifically the FOD-LBP and the SOD-LBP. The method for LBP FG/BG segmentation is presented in section 3 of which the results are presented in section 4. The FOD-LBP and the SOD-LBP are evaluated for edge detection and FG/BG segmentation in subsections 4.1 and 4.2, respectively. Finally, we conclude our paper in 5.

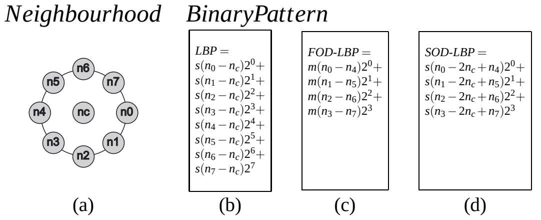

Figure 1: (a): A centre pixel and a circular neighbourhood of 8 pixels. (b) The basic LBP feature obtained by summing the thresholded differences weighted by powers of two. If the grey level of the neighbouring pixel is higher or equal, the value is set to one, otherwise to zero. (c): The FOD-LBP feature represents the magnitudes of the first order derivatives. (d): The SOD-LBP feature represents the signs of the second order derivatives.

2 EDGE-BASED LOCAL BINARY PATTERNS

In this section, we introduce edge-based LBPs with a novel generic strategy for higher order derivative LBPs. To understand the LBP concepts, we start this section by explaining basic LBPs.

2.1 Basic Local Binary Patterns

The basic LBP operator by Ojala et al. (Ojala et al., 2002) was proposed to describe local textural patterns. The LBP operator describes each pixel at position (x,y) by the relative grey levels of its neighbouring pixels as a binary number (binary pattern):

$$LBP_{R,N}(x,y) = \sum_{i=0}^{N-1} s(n_i - n_c)2^i,\ \ s(x) = \begin{cases} 1 & x \geq 0, \\ 0 & otherwise, \end{cases} \tag{1}$$

where n_c corresponds to the greylevel of the centre pixel of a local neighbourhood and n_i to the grey levels of N equally spaced pixels on a circle of radius R. The neighbouring pixel values are bilinearly interpolated whenever the sampling point is not in the centre of a pixel, as illustrated in Figure 1 (a). The pixels in the neighbourhood are thresholded by its centre pixel value, multiplied by powers of two and then summed to obtain a label for the centre pixel, as illustrated in Figure 1 (b). In practice, Eq. 1 means that the signs of the differences in a neighbourhood are interpreted as an N-bit binary number, resulting in 2^N distinct values for the binary pattern. The 2^N-bin histogram of the binary patterns computed over a region is used for texture description. The LBP features have proven to be robust against illumination changes, rotationally invariant, very fast to compute, and to not require many parameters being set (Ojala et al., 1996; Ojala et al., 2002).

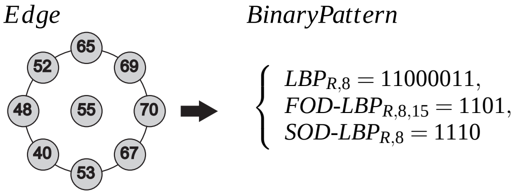

Figure 2: (a): A neighbourhood of 8 pixels representing an edge (b): Codes for basic LBP, FOD-LBP and SOD-LBP, respectively.

The histogram of the binary patterns computed over a region is used for texture description. To reduce the histogram size of the LBP, only so-called *uniform patterns* are considered (Mäenpää et al., 2000). To measure uniformity of a pattern, the number of bitwise transitions from 0 to 1 or vice versa is considered. An LBP is called uniform if its uniformity measure is at most 2. In uniform LBP mapping there is a separate output label for each uniform pattern and all the non-uniform patterns are assigned to a single label. Thus, the number of different output labels for mapping for patterns of N bits is $N(N-1)+3$. For instance, the uniform mapping produces 59 output labels for neighbourhoods of 8 sampling points. Uniform LBP codes represent local primitives including spots, flat areas, edges, edge ends, curves and so on. A numerical example of computing the basic LBP code of an edge pixel is shown in Figures 2. Note that the basic LBP represents the first-order circular derivative pattern of images, a micropattern generated by the concatenation of the binary gradient directions, as was shown in (Ahonen and Pietikäinen, 2009).

2.2 Higher Order Derivative Local Binary Patterns

Whereas the basic LBP operator is mainly used for texture analysis, in this work we employ the LBP as an edge operator. Therefore, we translate basic edge detection concepts to a novel generic LBP strategy. We introduce two novel LBP operators, i.e. the FOD-LBP operator and the SOD-LBP operator.

Edge operators (Gonzalez and Woods, 2001), such as the Sobel operator, use the magnitude of the first derivative (the gradient) to detect the presence of an edge at a point in an image (i.e., to determine if a point is on a ramp). Similarly, the sign of the second derivative (the Laplacian) is used to determine whether a pixel lies on a dark or light side of an edge. Definitions and discrete patterns of the 1-D gradient G_x and the 1-D Laplacian L_x of a function $g(x)$ are given in *gradient* *Pattern*

$G_x = \frac{\partial g(x)}{\partial x} \cong g(x) - g(x-1)$ $\boxed{1}\boxed{-1}$

Laplacian

$L_x = \frac{\partial^2 g(x)}{\partial^2 x} \cong g(x+1) - 2g(x) + g(x-1)$ $\boxed{1}\boxed{-2}\boxed{1}$

(a) (b)

Figure 3: $(a)-(b)$: Definitions and discrete patterns of the 1-D gradient G_x and the 1-D Laplacian L_x of a function $g(x)$.

Figure 3.

Basic LBPs as in Section 2.1 can be seen as patterns of the signs of the gradients, because like some gradient operators, it considers grey level differences between pairs of pixels in a neighbourhood. Here, the discrete pattern $\{1,-1\}$ of the gradient is translated into the FOD-LBP, which represents the magnitudes of the first order derivatives at position (x,y) as follows:

$$FOD\text{-}LBP_{R,N,T}(x,y) = \sum_{i=0}^{N/2-1} m(n_i - n_{i+N/2})2^i,$$
$$m(x) = \begin{cases} 1 & |x| \geq T_l, \\ 0 & otherwise, \end{cases} \tag{2}$$

where n_i and $n_{i+N/2}$ correspond to the grey values of centre-symmetric pairs of pixels of N equally spaced pixels on a circle of radius R. The absolute values of the grey level differences are thresholded with a small value T_l. A typical value for T_l is 10. In FOD-LBP, pixel values are not compared to the centre pixel but to the opposing pixel symmetrically with respect to the centre pixel, which is also illustrated in Figures 1 (a) and (c). This halves the number of comparisons for the same number of neighbours. We can see that for eight neighbours, FOD-LBP produces 16 different binary patterns. This modified scheme of comparing the pixels in the neighbourhood was already proposed in the form of Centre Symmetric LBPs (CS-LBPs) in (Heikkilä et al., 2009) as a solution to produce shorter histograms in the context of a region descriptor. Also, CS-LBP obtained robustness on flat image regions by thresholding the grey level differences at a typically non-zero threshold. FOD-LBP differs from CS-LBP in the evaluation function $m(x)$. FOD-LBP thresholds the absolute value of x, where CS-LBP thresholds the real value of x. A numerical example of computing the FOD-LBP code of an edge pixel is shown in Figure 2.

Similar to the FOD-LBP, the SOD-LBP represents the signs of the second order derivatives at position (x,y) as follows:

$$SOD\text{-}LBP_{R,N}(x,y) = \sum_{i=0}^{N/2-1} f(n_i - 2n_c + n_{i+N/2})2^i,$$
$$s(x) = \begin{cases} 1 & x \geq 0, \\ 0 & otherwise, \end{cases} \tag{3}$$

which is a translation of the discrete pattern $\{1,-2,1\}$ of the Laplacian operator to binary patterns, which is also illustrated in Figures 1 (a) and (d). A numerical example of computing the SOD-LBP code of an edge pixel is shown in Figure 2. The second order derivative is sensitive to noise. Therefore, for SOD-LBP only, we firstly pre-process the image with Gaussian smoothing as in the Laplacian of Gaussian method (Marr and Hildreth, 2000). The SOD-LBP looks for zero-crossings. Zero-crossings are places in the Laplacian of an image where the value of the Laplacian passes through zero, i.e. points where the Laplacian changes sign. Such points often occur at edges in images, i.e. points where the intensity of the image changes rapidly, but they also occur at places that are not as easy to associate with edges, so-called *false edges*. Therefore, we consider the SOD-LBP as some sort of feature detector rather than a specific edge detector.

As it was originally proposed in (Ojala et al., 1996), the practical implementation of the FOD-LBP operator and the SOD-LBP operator consider a 3x3 squared pixel block for each pixel of a low-resolution image to avoid floating point operations on the smart monochromatic sensors. Thus, for low-resolution images, the radius R is kept small, i.e. 1 pixel. Also in this work, the FOD-LBP operator and the SOD-LBP operator each assign different output labels to 14 uniform patterns and one single label to 2 non-uniform patterns. Remember that each operator produces 2^4 different binary patterns. Note that the FOD-LBP operator and the SOD-LBP operator are straightforwardly extendable to higher order derivative LBPs, e.g. the third order derivative LBP according to the discrete pattern $\{1,-3,3,-1\}$ in two circles of a 3x3 and a 5x5 neighbourhood, respectively.

3 FOREGROUND DETECTION FROM MOVING EDGES

FG/BG segmentation in this work describes the neighbourhood of each pixel in the background as a group of local binary pattern histograms. The differences with the framework proposed in (Heikkilä and Pietikäinen, 2006) are in the input and the binning of the histograms, yet the BG model update procedure is the same.

As input, we use edge-based uniform LBP features instead of texture-based LBP features, i.e. two 4-bit uniform binary codes of the FOD-LBP and the SOD-LBP instead of one 6-bit binary code of the basic LBP. Histogram binning is performed for each pixel by mapping 15 FOD-LBP labels and 15 SOD-LBP labels over two 6-bin histograms, respectively. The histograms contain 5 bins for the uniform labels and 1 bin for the non-uniform labels. Each bin covers a class of uniform labels which are rotated versions of one another, since rotated versions represent the same edge features (Ojala et al., 2002). Histogram binning in this work has the advantage of producing dense histograms with non-empty bins. In comparison, histogram binning in (Heikkilä and Pietikäinen, 2006) maps 64 output labels over a 64-bin histogram for each pixel, as such that the histograms suffer from sparsity.

In the following, we shortly explain the BG model update procedure as described in (Heikkilä and Pietikäinen, 2006). The evolution of the feature vectors of a particular pixel over time is considered as a pixel process. The LBP histogram computed over a circular region of a user-settable radius around the pixel is used as the feature vector. For low-resolution images, this radius is kept small (e.g. 1). The BG model for the pixel consists of a user-settable number of adaptive LBP histograms (e.g. 3). Each model histogram has a weight between 0 and 1 so that the weights sum up to one. The LBP histogram $\vec{h}$ of the given pixel from the new frame is compared to the current model histograms using the histogram intersection as a proximity measure. The threshold for the proximity measure, T_p, is a user-settable parameter (e.g. 0.7). If the proximity is below the threshold T_p for all model histograms, the model histogram with the lowest weight is replaced by $\vec{h}$ with an initial weight.

If matches were found, more processing is required. The best match is selected as the model histogram with the highest proximity value. The best matching model histogram is adapted with the new data by updating its bins using a user-settable learning rate (e.g. 0.01). The weights of the model histograms are updated with another user-settable learning rate (e.g. 0.01).

FG detection is done before updating the BG model. The histogram $\vec{h}$ is compared against the current BG histograms using the proximity measure as defined in the update algorithm. If the proximity is higher than the threshold T_p, the pixel is classified as BG. Otherwise, the pixel is marked as FG.

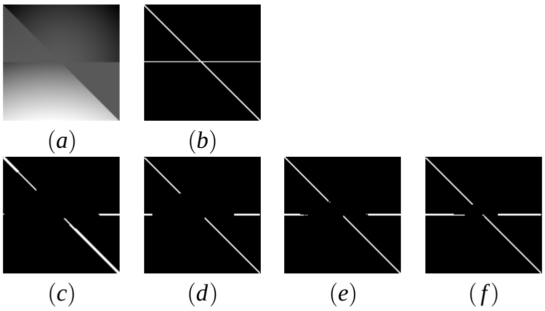
(a) (b)
(c) (d) (e) (f)

Figure 4: (a): An artificial greyscale image with an image resolution of 128x128 to evaluate the FOD-LBP edge operator and the SOD-LBP edge operator. (b): The ground truth edge image. $(c)-(d)$: Result of edge detection with the gradient operator, the FOD-LBP operator, the Laplacian operator and the SOD-LBP operator, respectively.

4 RESULTS

In the next two subsections, we firstly evaluate the FOD-LBP operator and the SOD-LBP operator for edge detection. Next, we compare the results of the proposed LBP FG/BG segmentation method with the results of the state-of-the-art technique in (Heikkilä and Pietikäinen, 2006).

4.1 Evaluation of LBP Edge Operators

To obtain the edge detection performance, the FOD-LBP operator and the SOD-LBP operator are evaluated on a subset of 20 meaningful artificial images with an image resolution of 128x128, of which Figure 4 (a) shows an example. The results are compared against ground truth, as in Figure 4 (b). We also compare the FOD-LBP operator and the SOD-LBP operator to the traditionally used gradient edge operator and Laplacian edge operator. The FOD-LBP operator and the gradient operator classify edge pixels with a threshold on the magnitude. The SOD-LBP operator and the Laplacian operator classify edge pixels from sign changes (zero-crossings). Additionally, the FOD-LBP operator and the SOD-LBP operator each classify pixels as edge pixels if the pattern is uniform with at least one bitwise transition. Figures 4 (c) to (f) show the results of edge detection with the gradient operator, the FOD-LBP operator, the Laplacian operator and the SOD-LBP operator, respectively.

Table 1 presents an overview of the True Positive Rates (TPR=TP/(TP+FN)) and the False Positive Rates (FPR=FP/(FP+TN)) for edge detection, where TP, FN, FP and TN are the amount of True Positives, False Negatives, False Positives and True Negatives, respectively. In our evaluation, we consider different levels of Gaussian noise corruption, where σ is the Gaussian kernel standard deviation. From the visual and numeric results we can conclude that the FOD-LBP operator and the SOD-LBP operator perform equally well or better than the gradient operator and the Laplacian operator, respectively. The better performance with the LBP approach is due to the larger amount of information about intensity differences. Another conclusion is that the Laplacian operator and the SOD-LBP operator are more sensitive to noise. Consequently, we firstly pre-process the image with Gaussian smoothing before FG/BG segmentation.

4.2 Evaluation of Edge-based LBP Foreground Segmentation

Low-resolution video in this work is obtained from a low-resolution sensor network. The sensor network consists of low resolution monochromatic smart sensors, so-called *mouse sensors*[1]. Each mouse sensor is controlled by a digital signal controller[2]. Algorithms run in an embedded memory constrained environment. Figure 5 *a* shows an example of the low resolution sensor controlled by the digital signal controller on a printed circuit board. Figure 5 *b* shows an example image with an image resolution of 30x30 pixels and an image depth of 6 bit. Video sequences are captured with illumination changes from fluorescent lamps and daylight in a room with a table, chairs and a window. The results of FG/BG segmentation are compared against ground truth. An example of a mouse sensor image and its ground truth FG mask are shown in Figures 6 (a) and (b), respectively. In the following experiments, we evaluate 150 frames spread over 10000 frames.

We compare the results of FG/BG segmentation produced by the FOD-LBP operator and the SOD-LBP operator with the results produced by the method described in (Heikkilä and Pietikäinen, 2006), which we denote as LBP. Figure 7 (a) shows a mouse sensor frame which we compute the FG for. Figures 7 (b), (c) and (d) show the FG masks of FG/BG segmentation when the BG is modelled with LBP features, FOD-LBP features and FOD-LBP + SOD-LBP features, respectively. In comparison with the texture-based LBP, we can clearly observe that the results with the FOD-LBP and the FOD-LBP + SOD-LBP

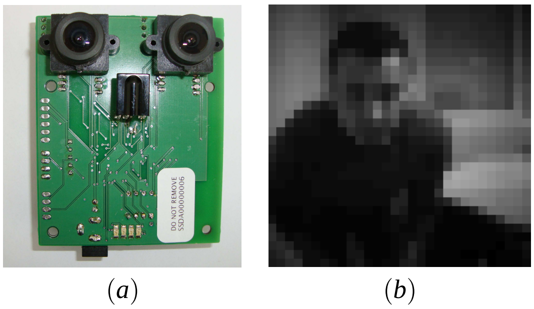

Figure 5: (a): A low resolution stereo-vision monochromatic smart sensor, a so-called *mouse sensor*. (b): An example image with an image resolution of 30x30 pixels and an image depth of 6 bit.

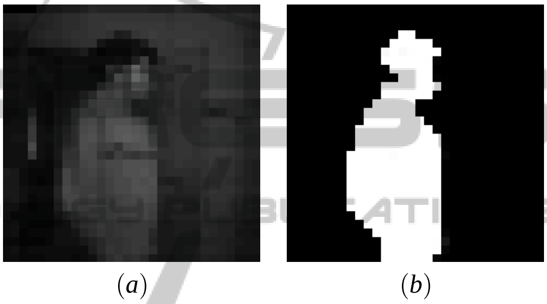

Figure 6: $(a)-(b)$: An example of a mouse sensor image and its ground truth.

extract the silhouette of the person. This is because the LBP technique considers FG as moving texture, whereas the FOD-LBP technique and the FOD-LBP + SOD-LBP technique consider FG as moving edges. The graph in Figure 8 plots the ROC curves (TPR versus FPR) for FG/BG segmentation with the LBP features, the FOD-LBP features and the FOD-LBP + SOD-LBP features. The ROC curves are obtained by a varying threshold T_p. When evaluated against ground truth, we conclude that the FOD-LBP technique and the FOD-LBP + SOD-LBP technique outperform the LBP technique. Using the FOD-LBP + SOD-LBP features is even better than using the FOD-LBP features only, since the FOD-LBP features only contain the magnitudes of the first order derivatives, while the FOD-LBP + SOD-LBP features also contain the signs of the second order derivatives, and thus more comprehensive edge information. The edge-based FG segmentation outperforms the texture-based method at low resolutions. Since texture information corresponds to 2-d area and edge information corresponds to 1-d contours, lowering the image resolution quadratically reduces texture information and linearly reduces edge information. Consequently, moving intensity differences are more relevant in low-resolution video.

In a second experiment, we consider video with a

[1] Agilent ADNS-3060: a high-performance optical mouse sensor with a programmable frame rate over 6400 frames, capturing images with an image resolution of 30*x*30 pixels and an image depth of 6 bit.

[2] Microchip dsPIC33FJ128GP802: a digital signal controller with a 16-bit wide data path and 64kB RAM.

Table 1: TPR and FPR of edge detection with the gradient operator, the FOD-LBP operator, the Laplacian operator and the SOD-LBP operator, when testing artificial images for different levels of Gaussian noise corruption. The FOD-LBP operator and the SOD-LBP operator perform equally well or better than the gradient operator and the Laplacian operator, respectively.

	Artificial images							
	gradient		FOD-LBP		Laplacian		SOD-LBP	
	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
$\sigma = 0$	0.817	0.010	**0.864**	**0.004**	0.927	0.005	**0.942**	**0.005**
$\sigma = 5$	0.833	0.010	0.864	0.004	0.924	0.007	0.945	0.104
$\sigma = 10$	0.834	0.010	0.866	0.004	0.921	0.082	0.953	0.378
$\sigma = 20$	0.830	0.010	0.862	0.006	0.926	0.351	0.977	0.746
$\sigma = 50$	0.827	0.023	0.892	0.242	0.920	0.702	0.985	0.943

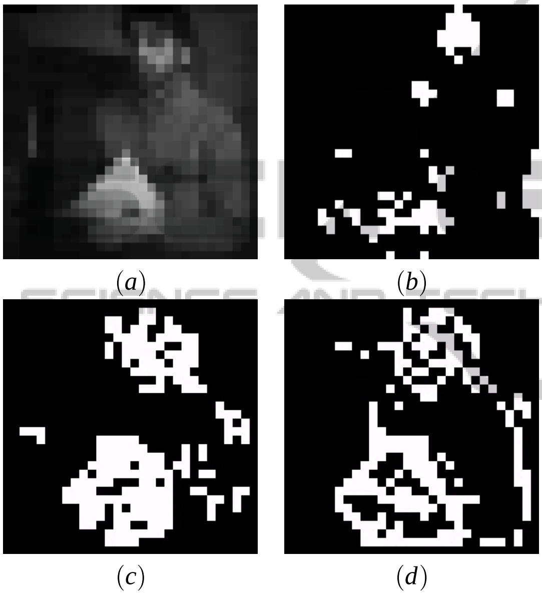
(a) (b) (c) (d)

Figure 7: (a): A mouse sensor frame with resolution of 30x30 pixels. $(b)-(d)$: The FG masks of FG/BG segmentation when the BG is modelled with the LBP features, the FOD-LBP features and the FOD-LBP + SOD-LBP features, respectively. The results with the FOD-LBP features and the FOD-LBP + SOD-LBP features extract the silhouette of the person.

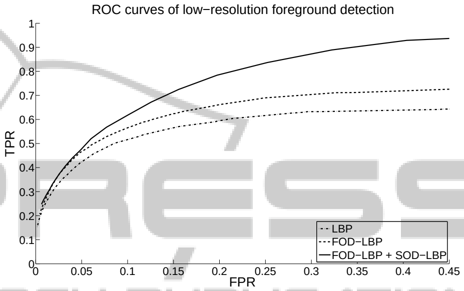

Figure 8: ROC curves for FG/BG segmentation in low-resolution video. Edge-based FG detection with the FOD-LBP operator and the SOD-LBP operator outperforms texture-based FG detection with the LBP operator (Heikkilä and Pietikäinen, 2006), since moving intensity differences are more visible in low-resolution video.

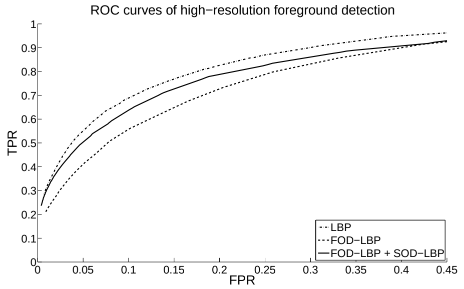

Figure 9: ROC curves for FG/BG segmentation in high-resolution video. Texture-based FG detection with LBP (Heikkilä and Pietikäinen, 2006) is better than edge-based FG detection with the FOD-LBP features and the FOD-LBP + SOD-LBP features, since texture becomes a more important factor in high-resolution video.

higher resolution of 384x288 pixels from the TUM Kitchen Data Set (Tenorth et al., 2009). From the ROC curves in Figure 9, we can conclude that the method with the LBP features has a better performance than the methods with the FOD-LBP features and the FOD-LBP + SOD-LBP features. This is because texture information becomes a more important factor in high-resolution video.

5 CONCLUSIONS

In this work, we presented extremely low-resolution foreground/background segmentation on video with resolution of 30x30 pixels. Therefore, we adapted a texture-based foreground segmentation algorithm based on LBPs into an edge-based method for low-resolution video processing. Edge information in the background model is introduced by a novel LBP strategy with higher order derivatives. Like the gradient operator and the Laplacian operator, edge information in this work is obtained by the magnitudes of FOD-LBPs and the signs of SOD-LBPs. In the results, foreground corresponds to edges on moving objects. The method is implemented and tested on low-resolution images produced by monochromatic smart sensors. The edge-based method outperforms texture-based foreground segmentation at low resolutions. In this work, we demonstrated that edge information becomes more relevant than texture information when the image resolution scales down.

ACKNOWLEDGEMENTS

The work was financially supported by iMinds and IWT through the Project 'LittleSister'.

REFERENCES

Ahonen, T., Hadid, A., and Pietikäinen, M. (2006). Face description with local binary patterns: application to face recognition. *IEEE Trans. on Pattern Recogn. and Machine Intelligence*, 28(12):2037–2041.

Ahonen, T. and Pietikäinen, M. (2009). Image description using joint distribution of filter bank responses. *Pattern Recognition Letters*, 30(4):368–376.

Barnich, O. and Droogenbroeck, M. V. (2009). Vibe: a powerful random technique to estimate the background in video sequences. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pages 945–948.

Camilli, M. and Kleihorst, R. P. (2011). Demo: Mouse sensor networks, the smart camera. In *ACM/IEEE Int. Conf. on Distributed Smart Cameras*, pages 1–3.

Gonzalez, R. C. and Woods, R. E. (2001). *Digital Image Processing*. Addison-Wesley Longman Publishing Co., Boston, MA, USA, 2nd edition.

Grüenwedel, S., Hese, P. V., and Philips, W. (2011). An edge-based approach for robust foreground detection. In *Proc. of Advanced Concepts for Intelligent Vision Systems*, pages 554–565.

Grünwedel, S., Jelača, V., Hese, P. V., Kleihorst, R., and Philips, W. (2011). Phd forum : Multi-view occupancy maps using a network of low resolution visual sensors. In *2011 Fifth ACM/IEEE Int. Conf. on Distributed Smart Cameras*. IEEE.

Grünwedel, S., Jelača, V., no Castañeda, J. N., Hese, P. V., Cauwelaert, D. V., Haerenborgh, D. V., Veelaert, P., and Philips, W. (2013). Low-complexity scalable distributed multi-camera tracking of humans. *ACM Transactions on Sensor Networks*, 10(2).

Heikkilä, M. and Pietikäinen, M. (2006). A texture-based method for modeling the background and detecting moving objects. *IEEE Trans. on Pattern Recognition and Machine Intelligence*, 28(4):657–662.

Heikkilä, M., Pietikäinen, M., and Schmid, C. (2009). Detection of interest regions with local binary patterns. *Pattern Recognition*, 42(3):425–436.

Hengstler, S. and Aghajan, H. (2006). A smart camera mote architecture for distributed intelligent surveillance. In *ACM SenSys Workshop on Distributed Smart Cameras*.

Huang, X., Li, S., and Wang, Y. (2004). Shape localization based on statistical method using extended local binary pattern. In *Proc. of Int. Conf. on Image and Graphics*, pages 184–187.

Mäenpää, T., Ojala, T., Pietikäinen, M., and Soriano, M. (2000). Robust texture classification by subsets of local binary patterns. In *Proc. of Int. Conf. on Pattern Recognition*, pages 935–938.

Marr, D. and Hildreth, E. (2000). Theory of edge detection. In *Proc. of Int. Conf. on Pattern Recognition*, volume 207, pages 935–938.

Movshon, J., Adelson, E. H., Gizzi, M. S., and Newsome, W. T. (1986). The analysis of moving visual patterns. *Pattern Recognition Mechanisms*, 54:117–151.

Ojala, T., Pietikäinen, M., and Harwood, D. (1996). A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29(1):51–59.

Ojala, T., Pietikäinen, M., and Mäenpää, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. on Pattern Recognition and Machine Intelligence*, 24(7):971–987.

Pietikäinen, M., Nurmela, T., Mäenpää, T., and Turtinen, M. (2004). View-based recognition of real-world textures. *Pattern Recognition*, 37(2):313–323.

Tenorth, M., , M., Bandouch, J., and Beetz, M. (2009). The tum kitchen data set of everyday manipulation activities for motion tracking and action recognition. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 1089–1096.

Yao, C.-H. and Chen, S.-Y. (2003). Retrieval of translated, rotated and scaled color textures. *Pattern Recognition*, 36(4):913–929.

Zhang, B., Gao, Y., Zhao, S., and Liu, J. (2010). Local derivative pattern versus local binary pattern: face recognition with high-order local pattern descriptor. *IEEE Trans. on Im. Proc.*, 19(2):533–544.

Zhao, S., Gao, Y., and Zhang, B. (2008). Sobel-lbp. In *IEEE Int. Conf on Image Processing*, pages 2144–2147.

Zivkovic, Z. (2004). Improved adaptive gaussian mixture model for background subtraction. In *Proc. of Int. Conf. on Pattern Recogn.*, pages 28–31.