Model of Human Clothes based on Saliency Maps

Sebastian Hommel, Darius Malysiak and Uwe Handmann
Computer Science Institute
University of Applied Sciences Ruhr West
Germany, Bottrop
E-Mail: (Sebastian.Hommel|Darius.Malysiak|Uwe.Handmann)@hs-ruhrwest.de

***Abstract*—In this paper, we describe a method to model human clothes for a later recognition by the use of RGB- and SWIR-cameras. A basic model is estimated during people detection and tracking. This model will be refined if the recognition is triggered. For the refining, several saliency maps are used to extract individual features. These individual features are located separately for any human body parts. The body parts are estimated by the use of a silhouette extraction combined with a skeleton estimation. In this way, the model describes the human clothes in a compact manner which allows the use of a simple and fast comparison method for people recognition. Such models can be used in security and service applications.**

***Index Terms*—people recognition, person model, clothing model, saliency maps, people detection, people tracking, silhouette extraction, skeleton estimation, illumination correction, security system, service application**

I. Introduction

The recognition of people is a everlasting topic in security applications and will become increasingly important in service applications to recognize interaction partners [1]. By the use of cameras, the face is commonly used to recognize people [2]. This method allows a recognition over a long time and a manipulation of this method is very hard. However, in a typical CCTV-System[1] the usage of facial features is often not possible, as the people don't necessarily look into the cameras. Additionally the resolution of a face image can be inadequately low for large distance observations. To make a people recognition possible in such situations, the human gait can be used as a biometric feature [3] or the features of a persons clothing can be utilized. The gait resembles an individual characteristic as the human face, but it differs massively by different ground conditions and different walking speeds. For that reason, we focus on a model of human clothes for the recognition of people over a longer period of time (i.e. a few hours). In another work regarding the modeling of human clothes, Hahnel [4] compares different methods to describe the color and texture of the clothing. A Kernel based method to compare simple features of human clothes for people recognition is presented in [5]. Takeuchi shows an improved PCA method for an automatic feature extraction [6]. An online feature selection method for a ranking-based recognition is visualized in [7]. In the work of Eisenbach, low level features are selected by comparing several features of different current known people, while our method selects conspicuous features separately for each person.

To test our model we use a currently being developed security application [8]. This application focuses on an intelligent support of video surveillance personal at airports. The decentralized system helps security staff to locate suspicious people within recorded videos, as well as to find their current position and estimate their next location. For each camera, a decentralized computer cluster indexes the videos during the recording. This process includes an illumination correction, people detection/tracking as well as a first feature extraction. A second centralized part is used to locate a marked person by the use of a geometric model, a face recognition and a whole-body recognition.

In this paper we will first describe our used camera types including their advantages and disadvantages as well as the applied illumination correction. In section III we will present methods for realtime people detection/tracking within high resolution videos. To locate features of human clothes for each body part[2], an estimation of these body parts is necessary. For this reason, a silhouette extraction and a skeleton estimation will be described in section IV. The main parts of this work, the feature extraction and the combination to a model of human clothing are explained in section V.

II. Types of Cameras

For this work two types of cameras have been used. Firstly, high resolution RGB-cameras[3] (1600*1200px) for the detection and tracking of people at great distances and extraction of fine individual clothing features. These cameras operate in the range of visible light (380nm to 780nm). The second type of cameras are short-wave-infrared-cameras (SWIR-cameras)[4] with a resolution of 320*256px. This kind of cameras records the short-wave part of infrared light. The operation range of the used InGaAs-sensor lies between 900nm and 1700nm. SWIR-cameras help visualizing different clothing material which is very helpful for so called business scenarios. In these scenarios a lot of people wear clothes which look similar within a RGB-image. However, on a SWIR-Image the clothing often differs massively. An example of a SWIR- and a corresponding RGB-image is shown in Fig. 1. A practical system uses many different cameras with a different kinds of lenses and

[1] Closed Circuit Television

[2] arms, torso, legs

[3] cameras with an red, green and blue channel

[4] SWIR-cameras are also called near-infrared-cameras (NIR-cameras)

978-1-4799-0197-5/13/$31.00 ©2013 IEEE

Fig. 1. **SWIR- and RGB-image** The SWIR-image (l) shows additional features for the recognition, which are not in the RGB-image (r).

resolutions. In this work we don't focus on any particular type of lenses or a specific camera resolution. Certainly, in this work the same type of lenses are used for one pair of cameras. A pair of cameras consists of one RGB- and one SWIR-camera, both are assumed to be mounted in a fixed position on a planar surface. In this way, the mapping of both camera images can be done by shifting and scaling.

A. Illumination Correction

To be able to extract clothing features with an invariance to different cameras, different daytimes, different weather conditions and lateral illumination, an illumination correction is necessary. For that reason, a sensor near image enhancement is calculated before the video stream will be recorded. The used image enhancement is based on the camera internal pixel representation (12bit per channel) which is higher than the cameras output (8bit per channel). In general, the internal bits are linearly mapped to the external image representation. It is preferable to use an adaptive logarithmic mapping function to correct different illuminations. In this work, the gamma correction [9] (Formula 1) is used.

$$f(x) = (x/2^n)^\gamma \cdot 255 \quad \text{; n = number of internal bits} \qquad (1)$$

By using this camera internal gamma correction, the image quantization noise does not increase since no sensor information is over-represented. The value of γ is estimated by Formula 2, with $min(dest)$ being the minimal destination value, $max(dest)$ the maximal destination value, $min(value)$ the minimal value of the current image and $max(value)$ the maximal possible value of the input image. As we mentioned before, in this work we map to an 8bit image, so $min(dest)$ is set to 1 and $max(dest)$ is set to 256. The current value is set to the mean of the image channels and the overall minimal value is searched for $min(value)$. In our implementation, the input image consists of 12bit for each channel, so $max(value)$ is set to 4096.

$$\gamma = \frac{log(min(dest)) - log(max(dest))}{log(min(value)) - log(max(value))} \qquad (2)$$

We limit γ to 0.63, since the use of a gamma correction with a significantly smaller γ leads to information loss (not every of the 8^2 values is usable in the external image representation). In the case of video analysis, it is preferable to smooth the gamma temporarily in order to handle short-time illumination changes (Formula 3).

$$\gamma_{t+1} = \gamma_t + \alpha \times (\gamma - \gamma_t) \quad \alpha \in [0, 1] \qquad (3)$$

This sensor near correction is used in combination with a low brightness threshold for a camera internal automatic exposure time adaptation. In this way, the recorded images are darker with less overexposure whereas the mapping function makes the image brighter and smoother in illumination. Thus the recorded 8bit image represents some of the previously overexposed image areas as well as the well-illuminated areas. With the help of this correction even previously black areas appear well illuminated (Figure 2). The decreasing of the threshold for the automatic exposure time adaptation results in a shorter illumination time which reduces the motion blur.

(a) without correction (b) with correction

Fig. 2. **Illumination correction** This figure shows an example result of the presented sensor near illumination correction. [8]

III. People Detection and Tracking

In order to generate a model of human clothes it is necessary to detect humans beforehand. To create a clothing model we combine the extracted features over time by two tracking methods.

A. People Detection

In order to reliably detect people within an image we utilized an algorithm known as *histogram of oriented gradients* (HOG [10]). The algorithms principle can be roughly summarized as follows. It extracts pixelwise edge-gradients from the input image and then assigns each gradient into one of nine orientation bins for a small (e.g. 8x8 pixel) image region. The orientation bins from each image region are then sequentially concatenated into a feature vector which in turn is being used as the input for a support vector machine (SVM) trained for people detection (we will refer to this as a HOG iteration). In order to increase the reliability for a successfull detection of humans we applied multiple support vector machines, each trained for a different feature set (e.g. head part, head-shoulder part etc.). Although we archieved a high rate of correct detections, the described approach also yielded long computation times. Processing a single frame on an ordanary CPU can take up to 10 seconds which shows the inherent drawback of the HOG. Thus we implemented a GPU based version of the detection algorithm and reduced the computation time to ≈ 90 milliseconds, utilizing multiple GPUs (one for each feature set) in parallel we finally removed this bottleneck in our framework of extracting features from human clothes. During our evaluation we dedicated a single computer to each camera, which enabled us to process the cameras data in realtime (i.e. 10 fps at 1600x1200), the systems structure is depicted on the left side of Fig. 3. This approach becomes unfeasible in realistic scenarios, e.g. multiple cameras with high framerates,

as we approach the physical limits of mainstream computers. Thus we developed a software framework to distribute the HOG in a cluster-like manor among small computation nodes, each equipped with one or two GPUs. The right side of Fig. 3 visualizes this approach, the systems structure follows the concept of a Beowulf cluster. In addition to the distributed

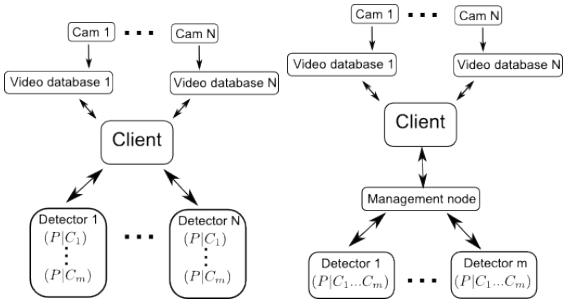

Fig. 3. **Structure of the detector system** The left image shows the use of multiple detectors, each executing m iterations $(P|C_i)$ of the HOG algorithm. The right image depicts an enchanced version of the same system. The main differences are: a) the workload is distributed via a management node among the detectors and b) the detector is using a modified HOG algorithm which preprocesses the image only once and uses multiple SVMs

computation we are planning to reduce the overhead within the parallel execution of HOG iterations. The HOG iterations within a single detector work on the same image data, thus the preprocessing P of the image data (i.e. extraction of the edge-gradients) can be executed only once. The classifiers C_i can then work on the same extracted feature set. This approach does not only allow a flexible system expansion for massive video data streams, it also provides reliable detection results using state of the art people detectors and allows even further preprocessing on each cluster nodes CPU (as the detection is entirely executed on the nodes GPU).

B. People Tracking

The tracking of detected people is necessary in order to group all the features of one person's clothes over time. To track the detected humans in a camera view, the tracker of Kolarow [11] is used, which is capable of generating long tracks of the detected areas. Afterwards a feature tracker is applied, which clusters similar general features (subsection V-A) to subtracks. This tracker is initialized with the general features of the first detection. Following this step, the initial features are searched in the next image close to the last location by using a Kalman filter without feature update. By using a small tolerance, only similar features are grouped to this subtracks.

IV. Estimation of Body Parts

To extract the body for the later feature extraction, a silhouette of the person is separated. Additionally to the silhouette, it is necessary to know where the body parts are located within the silhouette. For this task, a skeleton extraction is used.

A. Silhouette Extraction

There are a lot of possibilities to extract a silhouette. Firstly, it is possible to combine similar distances between sensor and objects to silhouettes. To calculate the distance, additional sensors like laser or IR-cameras could be used. Furthermore, it is possible to calculate the distance by the use of stereoscopic or moving cameras. Using fixed cameras, the silhouette can be extracted by the means of motion and foreground segmentation.

In this work, we use a foreground extraction since the usage of motion is only possible with moving objects. To make a fast extraction possible, a difference image method is used (Fig. 4). For our foreground extraction, the RGB-image is extended

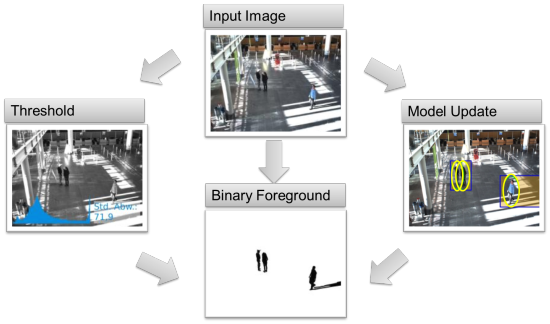

Fig. 4. **Silhouette extraction** First the threshold to evaluate the difference between input and background is calculated and a binary foreground mask is determined. Now, the background model is updated for non-person areas. [8]

with the SWIR-image to a RGBS-image by adding the SWIR-image as an additional channel. Firstly, a threshold to decide if a pixel is background or foreground is calculated by dividing the standard variance of the intensity image of the RGB-image by 4. This threshold is used for each channel of the RGBS-image. We also tried to use an individual threshold for each channel, which is determined by dividing the standard variance of the corresponding channel by 4. In our tests, this individual threshold was similar for each channel, yet no improvement regarding the resulting foreground map could be observed. A Pixel of the RGBS-image is only marked as a foreground element if the difference between the background and the input for each channel is greater than this threshold.

To accommodate variances of background illumination, each pixel (x, y) and each channel (c) of the background image (Ba) is adapted over time with an α of 0.125 (Formula 4).

$$\begin{aligned} Ba(x,y,c)_{t+1} =& Ba(x,y,c)_t \\ &+ \alpha * (RGBS(x,y,c) - Ba(x,y,c)_t) \end{aligned} \tag{4}$$

This adaption is connected with the result of the people detector which is described in subsection III-A. This combination allows an adaption of the background only in areas without people. In this way no people information is included into the background image.

B. Skeleton Estimation

The skeleton extraction from human silhouettes is the next step to estimate human body parts. For this work, the skeleton

extraction of Fan [12] is adapted. Fan maximized the radius of a circle stepwise while shiftling it inside a silhouette. Afterwards, new circles are initialized at the coordinates on the silhouette edge with the local maximum distances between the circles center and the silhouettes edge. All these circles are connected in the sequence of there generation. This connections are so called worms. The expansion of these worms is maximized under the restriction, that the worms remain inside the silhouette.

In this work, the upper part (size of silhouette divided by 2.5) of the silhouette will be eroded step by step until the silhouette image is empty. The first circle is initialized at the coordinates where a residual of the silhouette is, remain shortly before the image is empty. Firstly, the radius of this circle is maximized inside the silhouette without moving the circle. Next, the radius is maximized by moving the circle inside the silhouette. Hereafter, all distances (d, x, y) between the circles center and the silhouette edges are stored into a vector. This vector is divided in sequenced parts whom the distances are taller then $1.25 * mean\ distance$ to find the local maxima. In this work, two possibilities are treated, one local maximum or two local maxima. If only one local maximum is in a part of the vector, the next circle is initialized at the coordinates of this maximum distance. Though for two local maxima, the coordinates of the minimal distance between them is used. Fig. 5 exemplifies this process. In this work,

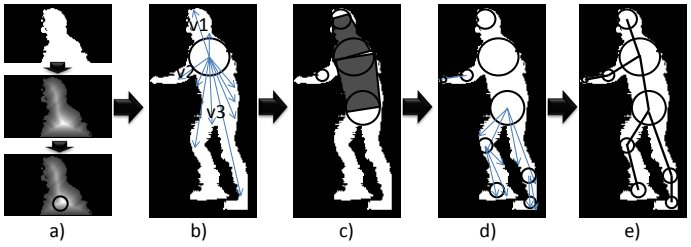

Fig. 5. **Skeleton extraction** a) initialization b) distance determination, grouping to distance vectors c) set new circles and trapezia d) next distances and circles e) connection of the circles (joints) to a skeleton

the worms are substituted by simple trapezia between the circles. Analogously to Fan, all the circles are connected in the sequence of their generation. In this way, the circles get parents and children. The resulted circles are associated with the body parts by simple rules:

- Is a child of the first circle on its left ⇒ this child and its children are parts of the left arm
- Is a child of the first circle on its right ⇒ this child and its children are parts of the right arm
- Is a child of the first circle on its top ⇒ this child and its children are parts of the head
- Is a child of the first circle on its bottom ⇒ this child is the hip
- Is a child of the hip on its left ⇒ this child and its children are parts of the left leg
- Is a child of the hip on its right ⇒ this child and its children are parts of the right leg
- The first circle and the hip are part of the torso

V. Feature Extraction

To speedup the recognition, two types of features will be used. The first kind are the *general features* of human clothes, which are calculated for each person during the tracking. These features are used to accelerate the search. In this way, the more complex *saliency maps based features* are only calculated for the searched person and a few hypotheses.

A. General Feature

The used general features are basically described in [1] for a human robot dialog system. Hommel categorizes appearance based features into color and texture features. The texture is naturally independent of the illumination. Whereas the RGB-color representation is transformed into the HSV-representation[5] (Formula 5) which is mostly independent in illumination since the value (V) describes the brightness. For the used color space, the hue (H) range from $0°$ to $360°$, the saturation (S) range from 0% to 100% as well as the value. Hommel used only the illumination independent hue and saturation of the color.

$$
\begin{aligned}
&R, G, B \in [0, 1] \\
&MAX = max(R, G, B); MIN = min(R, G, B) \\
&H = \begin{cases} 0°, & \text{if } MAX = MIN \\ 60° \cdot (0 + \frac{G-B}{MAX-MIN}), & \text{if } MAX = R \\ 60° \cdot (2 + \frac{B-R}{MAX-MIN}), & \text{if } MAX = G \\ 60° \cdot (4 + \frac{R-G}{MAX-MIN}), & \text{if } MAX = B \end{cases} \\
&H = H + 360°, \text{ if } H < 0° \\
&S = \begin{cases} 0, & \text{if } MAX = 0 \\ \frac{MAX-MIN}{MAX}, & \text{other} \end{cases} \\
&V = MAX
\end{aligned}
\tag{5}
$$

In this work, the HSV-Transformation is used to represent the color of the clothes which are recorded with the help of a RGB-camera. The used features are extracted on three fixed parts of the detected people (Fig. 6). One rectangle part of the lower body is separated to determine the mean hue and saturation from the RGB-image as well as the mean grey-scale value of the SWIR-image. Furthermore, the mean hue and saturation as well as the mean grey-scale value are calculated at a rectangle part of the upper body, too. At this rectangular upper body part, the mean horizontal and vertical texture rates are calculated for both, the RGB- and SWIR-image, with the help of the Scharr filter [13]. The mean horizontal and the mean vertical texture rates describe the strength of the texture at the selected area. One histogram of the hues and one histogram of the saturations are calculated from the rgb-image at an oval area of the upper body. At this area, one histogram of the grey-scale values is determined from the SWIR-image. The mean hue, saturation and grey-scale value describes the basic color and material of the users lower and upper body, while the histograms describes the upper body in

[5] hue, saturation, value

Fig. 6. **Feature extraction** The used features for the full body people recognition will be extracted from three areas. This areas are located relative to the people detection.

detail. By using the hue, saturation and grey-value histograms, even prints, patches etc. are represented in a very compact form. In this work, normalized histograms are used due to their scale independency. To handle minor changes, the hue and the saturation values as well as the grey-scale values are divided into 16 parts for the histograms.

As we described before, all the features will be tracked and integrated to similar subtracks over time in order to obtain a more robust and faster recognition. The features of each subtrack of a person track will be compared with all subtracks of the other tracks. For each track a certainty is set to the maximal similarity between the subtracks (Fig. 7). To compare

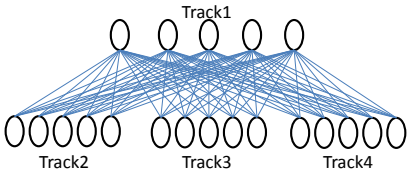

Fig. 7. **Track comparing** Each subtrack of Track1 are compared with each subtrack of all the other tracks.

the saved feature spaces, only the normalized differences of all the features are calculated and summarized. Certainly, during the calculation of the difference D between the hues of the searched person feature space (H_i) and the hue of the current hypotheses (H_c), one must heed that the hue is represented as a circle (Formula 6).

$$\begin{aligned} D &= |H_i - H_c| \\ D &= 360° - D, \text{ if } 360° - D < D \end{aligned} \tag{6}$$

In this way, the collection of each features differences is used as a score for each detection. Hence, a small score means a high similarity.

B. Saliency Maps based Features

Additional to the general features we utilize saliency maps in order to describe individual features of the human clothing. To find these features, saliency maps are calculated for each body part. These deformable body parts, are separated by using the silhouette extraction and skeleton estimation which we previously described. To locate comparable features, it is necessary to warp the body parts to standardized forms. For a faster extraction each body part is warped to a rectangle with a height of 100px. Afterwards, the features will be extracted separately for each body part. For this we combine the saliency maps of Itti and Koch [14] with the entropy of Kalinke [15] (Fig. 8). Firstly, a scale pyramid with 9 scales $I_0..I_8$ is

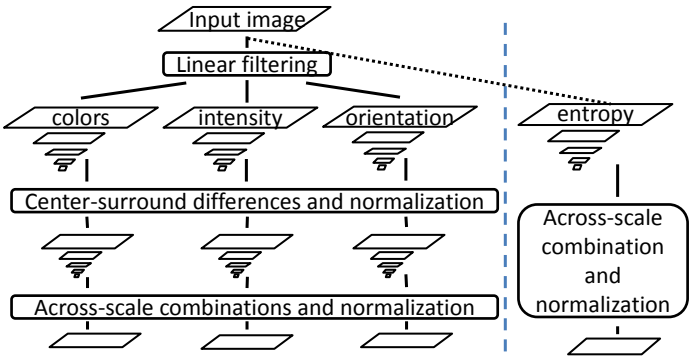

Fig. 8. **Saliency maps** Combination of the saliency maps of Itti and Koch (left) with the entropy of Kalinke (right).

calculated from a grey scaled image to calculate the intensity map I. All these 9 maps are scaled to the size of the fifth scale map (I_4) and will be combined (Formula 7).

$$I = \sum_{i=2}^{4} ((I_i - I_{i+3}) + (I_i - I_{i+4})) \tag{7}$$

After this combination, I is normalized and scaled to the input size. To calculate the orientation, the scale pyramid $I_0..I_8$ is also used. All the scale maps from I_2 to I_8 are filtered by cos-matrices for the angles $a \in (0°, 45°, 90°, 135°, 180°)$. The maps ($O_{i,a}$) are combined to an orientation map O (Formula 8).

$$O = \sum_{i=2}^{4} \sum_{a} ((O_{i,a} - O_{i+3,a}) + (O_{i,a} - O_{i+4,a})) \tag{8}$$

The map O is also normalized and scaled to the input size. A third saliency map, the so called color map which is based on two scale pyramids with 9 layers, is calculated from a red-green-image (RG) and a blue-yellow-image (BY). The RG is calculated by the red-channel (R), the green-channel (G) and the intensity image (I) (Formula 9)

$$RG = \frac{R - G}{I} \tag{9}$$

To calculate BY, the blue-channel (B) is additionally used (Formula 10).

$$BY = \frac{B - min(R, G)}{I} \tag{10}$$

The obtained 18 maps are scaled to the size of the fifth scale, too. These maps are used to calculate the color map (C) (Formula 11). For this calculation, an iterative normalization ($inor$) is used.

$$C = \sum_{i=2}^{4} \sum_{j=3}^{4} inor((BY_i - BY_{i+j}) + (RG_i - RG_{i+j})) \tag{11}$$

The obtained map C is also normalized and scaled to the input size.

Furthermore, the entropy map (E) is calculated with the help of the scales $I_0..I_8$ of the intensity image. Firstly, the entropy (E_i) is calculated separately for each intensity map. To calculate an entropy value for each pixel $e(x, y)$ (Formula 12), a local probability distribution is calculated for each pixel $p_k(x, y)$ $k \in (0, 1, 2, 3)$ (Formula 13). The value I_i at the position (x, y) of an intensity map ranges from 0 to 255.

$$e(x,y) = |\sum_{k=0}^{3} (p_k * \log_2 p_k)| \tag{12}$$

$$\begin{aligned} &p_k(x,y) = \frac{d_k(x,y)}{(width_{I_i}/16+1) * (height_{I_i}/16+1)} \\ &with \\ &\forall i \in (\lceil -width_{I_i}/32 \rceil .. \lceil width_{I_i}/32 \rceil) \; and \\ &\forall j \in (\lceil -height_{I_i}/32 \rceil .. \lceil height_{I_i}/32 \rceil) : \\ &d_{\lfloor I_i(x+i,y+j)/64 \rfloor}(x,y) = d_{\lfloor I_i(x+i,y+j)/64 \rfloor}(x,y) + 1 \end{aligned} \tag{13}$$

Afterwards, a threshold (t) for all pixel at the resulted entropy map is set to 1.25 times the mean entropy value of this map. All values of E_i which are smaller than or equal t are set to 0. Once all E_i are calculated, these maps are scaled to the input size. Next, all E_i are added to E which is normalized.

All these maps (color, intensity, orientation, entropy) are calculated for the RGB-image, certainly for the SWIR-image the color map is omitted. This results in seven saliency maps which are exemplary shown in Fig. 9. For all these maps, the

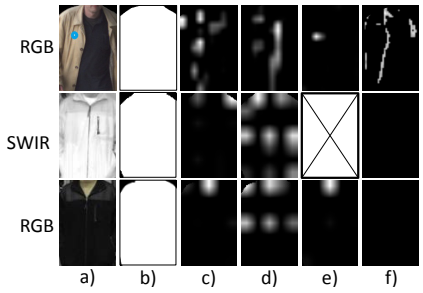

Fig. 9. **Saliency maps** a) input image b) body part mask c) intensity map d) orientation map e) color map f) entropy map

local maxima will be found. The location (x, y), the radius (r) and the value (v) of all these maxima are stored in a matrix M_i, separately for every body part. All these matrices are combined to matrix M.

With the help of the person tracks, these matrices are summarized over time. In this way, several views of one person are connected. To recognize one person, each matrix M in a track is compared with all matrices of the human detections from other possible tracks, similar to the method which is described in subsection V-A.

VI. Conclusions

In this paper, we presented a method to detect people and extract clothing features with respect to individual body parts by combining RGB- and SWIR-cameras. The extraction of general features can be done parallel to the detection process. Although it yields good detection results, our applied detection method exhibits a high time complexity. Yet it can be deployed in a massively parallel way. In order to utilize this attribute we described a beowulf cluster which can be flexibly expanded for practical systems (i.e. systems with many cameras). We showed that a fast and robust recognition can be realized by tracking these features and clustering them with respect to their similarity. A pre-selection of possible tracks according to these general features enables one to calculate saliency maps which can be used to enhance the recognition rate even further. Through the used people detection method, it is not possible to handle partially occlusion but this should be possible by using body part detectors which becomes tested in further work.

Acknowledgment

This work was partly funded by the German Federal Ministry of Education and Research (BMBF) in the framework of the APFel project.

References

[1] S. Hommel, A. Rabie, and U. Handmann, *Intelligent Systems: Models and Applications*, ser. Topics in Intelligent Engineering and Informatics. Springer Berlin Heidelberg, 2013, vol. 3, ch. Attention and Emotion Based Adaption of Dialog Systems, pp. 215–235.

[2] U. Handmann, S. Hommel, M. Brauckmann, and M. Dose, *Towards Service Robots for Everyday Environments*, ser. Springer Tracts in Advanced Robotics. Springer Berlin / Heidelberg, 2012, vol. 76, ch. Face Detection and Person Identification on Mobile Platforms, pp. 227–234.

[3] Y. Iwashita, R. Kurazume, and K. Ogawara, "Expanding gait identification methods from straight to curved trajectories," *WACV*, pp. 193–199, 2013.

[4] M. Hahnel, D. Klunder, and K.-F. Kraiss, "Color and texture features for person recognition," *IJCNN*, pp. 647–652, 2004.

[5] K. Yoon, D. Harwood, and L. S. Davis, "Appearance-based person recognition using color/path-length profile," *JVCIR*, vol. 17, pp. 605–622, 2006.

[6] Y. Takeuchi, M. Ito, K. Kashihara, and M. Fukumi, "Novel supervised feature extraction algorithm based on iterative calculations," *IRI*, pp. 304–308, 2011.

[7] M. Eisenbach, A. Kolarow, K. Schenk, K. Debes, and H.-M. Gross, "View invariant appearance-based person reidentification using fast online feature selection and score level fusion," *AVSS*, pp. 184–190, 2012.

[8] S. Hommel, M. A. Grimm, V. Voges, U. Handmann, and U. Weigmann, "An intelligent system architecture for multi-camera human tracking at airports," *CINTI*, pp. 175–180, 2012.

[9] J. Scott and M. Pusateri, "Towards real-time hardware gamma correction for dynamic contrast enhancement," *AIPR*, pp. 1–5, oct. 2009.

[10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *CVPR*, vol. 1, pp. 886–893, 2005.

[11] A. Kolarow, M. Brauckmann, M. Eisenbach, K. Schenk, E. Einhorn, K. Debes, and H.-M. Gross, "Vision-based hyper-real-time object tracker for human-robot interaction," *IROS*, 2012.

[12] B. Fan and Z.-F. Wang, "Pose estimation of human body based on silhouette images," *ICIA*, pp. 296–300, 2004.

[13] H. Scharr, *Optimal operators in digital image processing*. Ph.D. thesis, Interdisciplinary Center for Scientific Computer, Ruprecht-Karls-Universität, Heidelberg, 2000.

[14] L. Itti and C. Koch, "A saliency-based search mechanism for overt and covert shifts of visual attention," *VISRES*, vol. 40, no. 10-12, pp. 1489–1506, 2000.

[15] T. Kalinke and W. v. Seelen, "Entropie als Maß des lokalen Informationsgehalts in Bildern zur Realisierung einer Aufmerksamkeitssteuerung," *DAGM*, pp. 627–634, 1996.