Person Appearance Modeling and Orientation Estimation using Spherical Harmonics

M. C. Liem and D. M. Gavrila

Abstract— **We present a novel approach for the joint estimation of a person's overall body orientation, 3D shape and texture, from overlapping cameras. Overall body orientation (i.e. rotation around torso major axis) is estimated by minimizing the difference between a learned texture model in a canonical orientation and a texture sampled using the current 3D shape estimate (i.e. torso and head). The estimated body orientation subsequently allows to update the 3D shape estimate, taking into account the new 3D shape measurement obtained by volume carving. Our main contribution is a method for estimating a person's relative body orientation while simultaneously generating a basic Spherical Harmonics based model of the person's shape and texture.**

Experiments show that the proposed method outperforms two state-of-the-art orientation estimation methods: one combining a fixed 3D shape model with a generate-and-test texture matching approach and one using a classifier based approach.

I. Introduction

3D human pose is an important feature for human activity and social behavior analysis. As part of the 3D pose, a person's body orientation (i.e. rotation around 3D torso major axis) conveys much important information about the person's current activity, whether indicating the person's direction of focus, direction of movement or interaction level with other people in the scene.

In this paper, we focus on the estimation of overall person body orientation from few, overlapping cameras. To obtain a more accurate orientation estimate, we jointly estimate it with a 3D shape and texture representation of a person's torso-head, under a rigidity assumption. By projection onto a basis of Spherical Harmonics (*SH*), a low dimensional appearance model is created that can cope with object deformations while retaining the spatial information captured in a textured 3D representation. By comparing the texture model of the person at consecutive points in time, the relative body orientation can be estimated elegantly using the properties of the SH. This in turn allows to update the 3D shape and texture model of a person.

Apart from facilitating an accurate orientation estimate, the proposed 3D shape and texture model has furthermore the potential to offer improved track disambiguation in a multiple person scenario, compared to a simpler 2D based model. It could also provide an initialization for applications aiming at full articulated 3D pose recovery.

M. C. Liem and D. M. Gavrila are with the Intelligent Systems Laboratory, University of Amsterdam, 1098 XH Amsterdam, The Netherlands {M.C.Liem,D.M.Gavrila}@uva.nl

This research has received funding from the EC's Seventh Framework Programme under grant agreement number 218197, the ADABTS project.

II. Previous Work

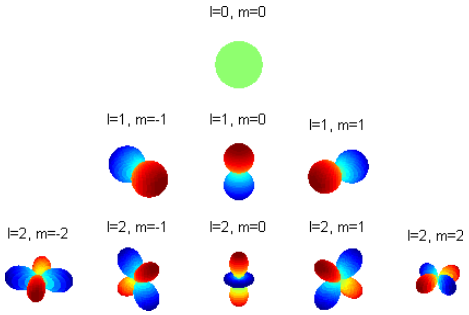
Estimating person orientation has been addressed in multiple ways. Several 2D single frame approaches combine orientation-specific person detectors [1], [2], [3]. Work by [4] estimates body and head pose in batch mode, coupling the output of underlying classifiers. In [5], a texture sampled from a cylinder surrounding the person is shifted along the rotational axis to find the best matching orientation and simultaneously form an appearance model.

Modeling person appearance is most commonly done based on single view information. Color histograms representing the full extent of a person's appearance are often used [6], while more sophisticated methods split a single person's appearance up into several layers, incorporating some spatial information in the descriptor [7], [8]. Some approaches also combine histograms created at multiple viewpoints [7].

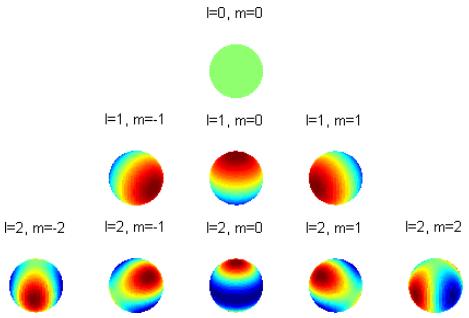
Since the human body shape is largely non-rigid, modeling body texture is not straightforward. One approach is to ignore the shape changes over time and map a full-body texture onto a rigid approximation of the body shape (e.g. a cylinder [5]). An alternative approach is to aim for full articulated 3D body pose and model estimation, with all the challenges involved (robustness, computational cost), as done in [9], [10], [11].

SH have been used in order to perform face recognition under varying lighting conditions [12]. Representing 3D objects by projecting them onto a *SH* basis has been researched mainly with respect to exemplar based 3D object retrieval. A collection of rotationally invariant 3D object descriptors has been compared in [13]. Recently, a *SH* decomposition of a 3D extension of the HOG descriptor was presented in [14].

In this paper, we take an intermediate approach between a fixed body shape and full articulated pose. We estimate the largest rigid partition (core) of the body over time by regarding all non-rigid elements of the human body (arms, legs) as 'noise' around the rigid body core (torso-head). Our main contribution is a method for estimating a person's relative body orientation while simultaneously generating a basic model of the person's shape and texture. The estimate is made on a single frame basis using an on-line learned appearance model consisting of low dimensional *SH* shape and texture representations, without the need for any a-priori shape or texture information. By using *SH* as a basis, orientation estimation can be performed elegantly, without the need of explicitly testing for different orientations. Furthermore, the low dimensional representation gives us an appearance model that is robust against local changes in shape or texture by implicitly smoothing the models.



(a) *SH* components for geometric deformation



(b) *SH* components for (color) gradient

Fig. 1. Spherical basis functions for the first three *SH* bands ($L = 2$). *SH* components can be visualized by representing spherical function f as a deformation of a sphere (a) or as a texture gradient (b).

III. Spherical Harmonics

In order to simplify the estimation of a person's orientation at time t based on a shape model $\mathcal{S}_t$ and texture model $\mathcal{T}_t$, we represent both in a Spherical Harmonic (*SH*) subspace. *SH* are the equivalent of Fourier transformations on a 3D sphere; they decompose a spherical function $f(u, v)$ (a function defined on the surface of a sphere) into a linear combination of orthonormal spherical basis functions. A *SH* subspace consists of L bands, and M components (spherical basis functions) per band, where for a certain band $0 \leq l \leq L$, $M = 2l + 1$. The *SH* decomposition of a spherical function f computes *SH* parameters $\mathcal{A}$, where $\mathcal{A}^{l,m}$ is the weight of component m in band l such that $SF_{\mathcal{A}}$ is the reconstruction of f from $\mathcal{A}$, thus $SF_{\mathcal{A}}(u, v) = f(u, v)$. In practice we use a limited number of bands such that $SF_{\mathcal{A}}$ is a band limited approximation of f.

If a 3D shape is represented by f_S as the protrusion of a sphere along vertices (u, v), then its *SH* decomposition can be seen as a linear combination of canonical shapes of which fig. 1(a) shows the first 9. Analogous, the *SH* decomposition of a texture represented by f_T as color values at vertices (u, v) can be seen as a linear combination of gradient maps seen in fig. 1(b). More details on *SH* and how to decompose a 3D object into *SH* components can be found in [15].

Equivalent to the translational invariance of Fourier transformations, *SH* are rotationally invariant. This means that any rotation of a 3D object described by *SH* components is the same as the rotation of its *SH* components. Because *SH* form an orthonormal basis, a rotation of an object projected onto a *SH* basis results in a shift of the components weights in each band [16]. Rotation of a 3D object along the vertical axis in *SH* is done according to the following function:

$$\mathcal{A}^{l,m} e^{-im\phi} = \mathcal{B}^{l,m}. \tag{1}$$

Here $\mathcal{A}^{l,m}$ is the weight of component m in band l before rotation, i is the imaginary unit and $\mathcal{B}^{l,m}$ is the weight of component m in band l after rotation over angle ϕ. For notational simplicity, we will sometimes leave out the superscript l, m when describing the rotation of a complete *SH* object and use $\mathcal{A}^{\phi}$ to denote $\mathcal{B}$. In those cases, rotation is assumed to be applied to all components l, m of $\mathcal{A}$.

In case a *SH* parametrization $\mathcal{A}$ and $\mathcal{B}$ of same object is given, the rotation ϕ between them can be found by minimizing the following sum squared error (SSE) using a quasi-Newton approach like BFGS [16]:

$$\sum_{1 \leq l \leq L} \sum_{1 \leq m \leq l} (\mathcal{A}^{l,m} e^{-im\phi} - \mathcal{B}^{l,m})^2. \tag{2}$$

Here, L is the number of bands selected for representing the object. The 0^{th} component of every band is not needed for this estimation since it is not influenced under rotation.

IV. 3D Appearance

At each time step t, C images $I_t^{1:C}$ are captured from C different, overlapping viewpoints. In our experiments, $C = 3$. Foreground estimation is done for each image and a volumetric reconstruction of the scene is created using volume carving. While there is only one person in the scene, all reconstructed volumes too small to represent a person are discarded. The remaining set of voxels is the person's visual hull $\mathcal{H}$. The person's measured position $\hat{x}_t$ is based on the ground plane projection of the center of the principal axis of $\mathcal{H}$. A constant acceleration Kalman filter is used to filter $\hat{x}_t$ over time and gives the filtered person position x_t.

In order to use *SH* to model the person's shape, $\mathcal{H}$ is transformed into spherical shape function f_S. To create f_S, a unit sphere is centered on the person location x_t and the sphere's surface is discretized into a uniformly distributed set of surface points (in our experiments we use 55×55 surface points). For each surface point (u, v), a ray r is cast from x_t, trough (u, v). The spherical function is defined by $f_S(u, v) = d$, where d is the distance between x_t and the most distant voxel in $\mathcal{H}$ along r. This is similar to the method described in [17] except that, in order to maintain rotational information, we do not normalize the 3D shape. To compensate for the fact that volume carving tends to over-estimate the shape, the spherical function is scaled down 10% to prevent sampling of the texture outside the object during texture generation.

A sampled texture consists of a spherical texture function f_T, created by projecting the surface points of a shape function f_S onto images $I_t^{1:C}$ and sampling the image values at these locations. Texture is only sampled from image regions containing foreground and the color of surface points visible in multiple cameras is averaged over these cameras.

The goal is to model the person's appearance, consisting of a *SH* shape model $\mathcal{S}_t$ and a *SH* texture model $\mathcal{T}_t$, and estimate

Algorithm 1: Appearance modeling and orientation estimation

Input: $\mathcal{S}_{t-1}$, $\mathcal{T}_{t-1}$, x_{t-1}, ϕ_{t-1}, $I_t^{1:C}$, $\mathcal{H}$
Output: $\mathcal{S}_t$, $\mathcal{T}_t$, x_t, ϕ_t

1 $\hat{x}_t$ = person position based on average voxel position of $\mathcal{H}$;
2 x_t^- = Kalman filter prediction of person position using x_{t-1};
3 x_t = Kalman filter update of x_t^- using measurement $\hat{x}_t$;
4 ϕ_t^- = Kalman filter prediction of orientation using ϕ_{t-1};
5 $\Delta\phi = \infty$; Intermediate orientation difference;
// Determine texture model $\mathcal{T}_t$ and orientation ϕ_t
6 **while** $\Delta\phi \geq 0.001$ **do**
7 **if** *not first iteration* **then** $\hat{\phi}_t = \hat{\phi}_t + \Delta\phi$;
8 **else** $\hat{\phi}_t = \phi_t^-$;
9 Rotate shape model: $\mathcal{S}_{t-1}^{\phi} = \mathcal{S}_{t-1} e^{im\hat{\phi}_t}$;
10 Reconstruct spherical function $SF_{\mathcal{S}^{\phi}}$ from $\mathcal{S}_{t-1}^{\phi}$;
11 Position $SF_{\mathcal{S}^{\phi}}$ at x_t;
12 Project surface points $SF_{\mathcal{S}^{\phi}}(u,v)$ onto $I_t^{1:C}$;
13 Sample texture from $I_t^{1:C}$ at projected surface points (i,j), creating $f_T(u,v) = \frac{1}{C}\sum_C I_t^c(i,j)$;
14 Compute *SH* representation $\hat{\mathcal{T}}_t$ of f_T;
15 $\Delta\phi = \arg\min_{\phi}[\sum_{\lambda}\sum_{l}\sum_{m}(\hat{\mathcal{T}}_t^{l,m,\lambda} e^{-im\hat{\phi}} - \mathcal{T}_{t-1}^{l,m,\lambda})^2]$;
16 ϕ_t = Kalman filter update of ϕ_t^- using measurement $\hat{\phi}_t$;
17 **if** $t < 1/\alpha_{\mathcal{T}}$ **then**
18 $\mathcal{T}_t = \frac{t-1}{t}\mathcal{T}_{t-1} + \frac{1}{t}\hat{\mathcal{T}}_t$;
19 **else**
20 $\mathcal{T}_t = (1-\alpha_{\mathcal{T}})\mathcal{T}_{t-1} + \alpha_{\mathcal{T}}\hat{\mathcal{T}}_t$;
// Compute shape model $\mathcal{S}_t$ based on ϕ_t
21 Position a unit sphere with 55×55 discretized surface points at x_t;
22 **foreach** *surface point* (u,v) **do**
23 Cast ray r from x_t trough (u,v);
24 Determine distance d between x_t and the voxel in $\mathcal{H}$ along r furthest away from x_t;
25 Compute scaled f_S using $f_S(u,v) = 0.9d$;
26 Compute the *SH* representation $\hat{\mathcal{S}}_t$ of f_S;
27 $\forall l,m \in \hat{\mathcal{S}}_t : \hat{\mathcal{S}}_t^{\phi} = \hat{\mathcal{S}}_t e^{-im\phi_t}$;
28 $\mathcal{S}_0 = \hat{\mathcal{S}}_0; \quad \mathcal{S}_t = (1-\alpha_{\mathcal{S}})\mathcal{S}_{t-1} + \alpha_{\mathcal{S}}\hat{\mathcal{S}}_t^{\phi}$;

the person's orientation ϕ_t (rotation along the vertical axis) based on the estimated appearance.

Algorithm 1 describes the *SH* based appearance modeling and orientation estimation method. The different parts of this method will be explained in this section.

A. Texture Model

First, a constant acceleration Kalman filter is used to get a prediction ϕ_t^- of person orientation ϕ_t based on ϕ_{t-1}. Using the shape model $\mathcal{S}_{t-1}$ from the previous timestep, the *SH* texture measurement $\hat{\mathcal{T}}_t$ at time t is computed and the person's orientation ϕ_t is estimated using this measurement.

Texture function f_T at t is created using a rotated spherical shape function $SF_{\mathcal{S}^{\phi}}$, reconstructed from the *SH* shape model $\mathcal{S}_{t-1}^{\phi}$ acquired by rotating $\mathcal{S}_{t-1}$ using the predicted orientation according to line 9 of alg. 1. When positioned at x_t, $SF_{\mathcal{S}^{\phi}}$'s discretized surface points can be used to sample the person's texture from images $I_t^{1:C}$. Texture regions without color information due to occlusions or sampling outside the foreground regions are filled using the average color along horizontal scanning lines over the texture. This way, artifacts in the *SH* texture representation due to lacking information are prevented while vertical color variance is maintained in the texture. $\hat{\mathcal{T}}_t$ is computed by projecting f_T onto a 9 band *SH* subspace ($L = 8$), reducing the dimensionality of the texture space is by 98% and smoothing the texture. To use RGB color, the three color channels $\lambda = \{R, G, B\}$ are modeled as three separate *SH*.

Since rotation of the *SH* texture is simplified according to (1), finding the most likely orientation $\hat{\phi}_t$ given the *SH* texture model from the previous timestep $\mathcal{T}_{t-1}$ and the current texture measurement $\hat{\mathcal{T}}_t$ is straightforward and can be solved by minimizing the SSE on line 15 of alg. 1 for $\hat{\phi}_t$ using a quasi-Newton approach like BFGS [16]. A Kalman filter update of ϕ_t^- using $\hat{\phi}_t$ gives the final orientation ϕ_t.

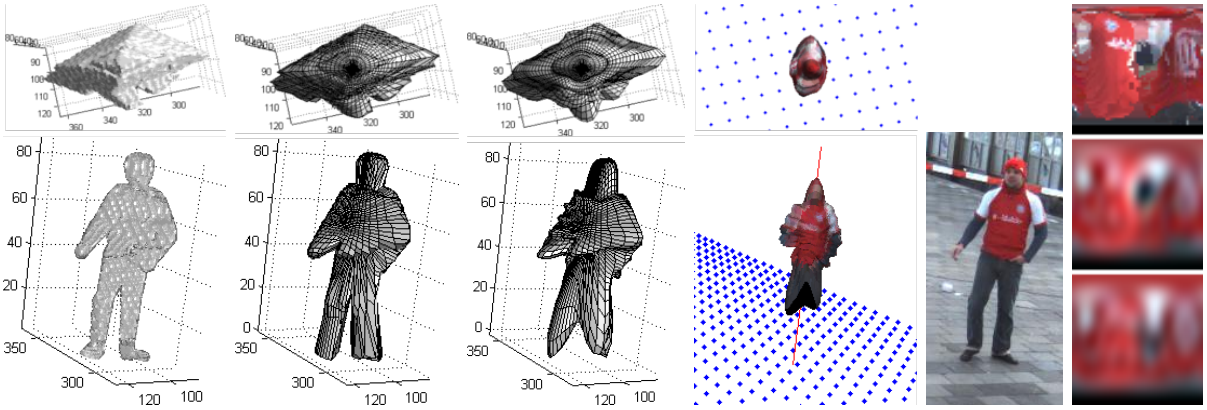
The time-based texture model $\mathcal{T}_t$ is learned by exponential decay using learning rate $\alpha_{\mathcal{T}}$ according to the equation on line 20 of alg. 1. However, to prevent overrepresentation of the first sampled textures in $\mathcal{T}_t$, iterative averaging is used to learn $\mathcal{T}_t$ as long as $t < \frac{1}{\alpha_{\mathcal{T}}}$, as shown on line 18 of alg. 1. Combining models and measurements over time requires each measurement to be rotated into a canonical orientation, matching the model. By sampling the texture using the rotated shape model $\mathcal{S}_{t-1}^{\phi}$, $\hat{\mathcal{T}}_t$ is in a canonical orientation and can directly be combined with $\mathcal{T}_{t-1}$. An example of a sampled texture, its *SH* reconstruction and a learned texture after 140 frames can be found in fig. 2(f).

B. 3D Shape Model

Using orientation ϕ_t, the *SH* shape model $\mathcal{S}_t$ can be computed. First, the *SH* shape measurement $\hat{\mathcal{S}}_t$ at time t is constructed by transforming visual hull $\mathcal{H}_t$ into a spherical function f_S and projecting this function onto the *SH* space. To reduce feature dimensionality and simultaneously smooth the shape representation, the first 17 bands of the *SH* space ($L = 16$) are used for projection, reducing feature dimensionality by 90%. Examples of a person's visual hull, the spherical function derived from that visual hull and the reconstruction of the *SH* representation of the spherical function can be seen in fig. 2(a)-(c). The top-down views showing the person facing to the right clearly show the excess volume created by volume carving due to shape ambiguities.

Estimating the rigid body over time is done by averaging shape estimates over time, decreasing the influence of non-rigid body parts on the estimated body shape. Since arms and legs will have different positions over time, they will be averaged out in the final shape estimate as can be seen in fig. 2(d). To combine shape models and measurements, they have to be in canonical orientation. Since the shape is represented in *SH* components, rotation is straightforward (as mentioned in sec. III) and does not yield rotation artifacts which would occur when rotating objects in the discrete volume space consisting of cubic voxels in a regular grid. The rotated *SH* shape measurement $\hat{\mathcal{S}}_t^{\phi}$ is computed as shown in alg. 1 on line 27, in accordance with (1).

The rigid body shape model $\mathcal{S}_t$ at time t is computed over time by iteratively combining $\hat{\mathcal{S}}_t^{\phi}$ for all timesteps under exponential decay, as shown in alg. 1 on line 28. The shape learning rate, $\alpha_{\mathcal{S}}$, is split up in two parts: one for growing the model and one for shrinking it. Because of the nature of volume carving, it is much more likely that carving artifacts consist of extrusions of the actual shape instead of indentations. To handle artifacts efficiently, the learning rate



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

Fig. 2. Examples taken from the person shown in fig. 3(c). (a)-(d) show a top-down view of the person facing to the right (top) as well as a camera perspective view (bottom). (a) Person visual hull $\mathcal{H}$. (b) Spherical shape function using 55×55 surface points. (c) 17 band *SH* shape representation. (d) Learned shape model after 140 frames, mapped with sampled texture. (e) Person modeled. (f) top-to-bottom: sampled texture (55×55 pixels), 9 band *SH* texture representation, learned *SH* texture after 140 frames.

for surface points that are more indented in $\hat{\mathcal{S}}_t^{\phi}$ than in $\mathcal{S}_{t-1}$ is set to 0.4. The learning rate for more extruded surface points is set to 0.01. An example of a learned shape model can be found in fig. 2(d), showing that the model learned a good approximation of the actual person's volume.

C. Iterative Orientation Estimation

The estimate of the current body orientation made using the equation on line 15 of alg. 1 allows for an iterative orientation optimization scheme. Since the texture measurement $\hat{\mathcal{T}}_t$ is sampled using the reconstruction of $\mathcal{S}_{t-1}$, oriented according to the Kalman filter prediction ϕ_t^- instead of the final estimate ϕ_t, the sampled texture might be distorted.

In order to optimize the estimation of the person orientation, texture sampling and orientation estimation are repeated multiple times. Each time, the reconstruction of $\mathcal{S}_{t-1}$ is rotated using the most likely orientation estimate $\hat{\phi}_t$ of the previous iteration. While the estimate gets closer to the true orientation, the SSE gets smaller and the difference between orientation estimates reduces. Optimization is stopped when the difference between consecutive orientation estimates is less than 0.001 rad or 10 iterations have been done.

$\mathcal{S}_{t-1}$ could be updated every iteration using $\hat{\mathcal{S}}_t^{\phi}$, rotated using the previous iteration's estimate of $\hat{\phi}_t$. However, while the orientation estimate is suboptimal, an incorrectly rotated version $\hat{\mathcal{S}}_t^{\phi}$ might result in a more distorted shape estimate instead of an improved estimate.

V. Experiments

We evaluated how the *SH* texture representation, shape information and the iterative estimation procedure influence the quality of the estimated orientation. To this end, our method was compared to two state of the art approaches. The first method is based on the Panoramic Appearance Map (PAM) [5]. A fixed size cylinder with a diameter of 35 cm and a height of 2 m, fitting tightly around a person's torso is used for sampling the person's texture. Orientation is estimated in a generate-and-test fashion by sampling 360 textures using 1° orientation difference and comparing all textures to a learned texture model. The orientation of the best matching texture sample is used as the object orientation. Alternatively, we combined the cylinder based shape model with our SH based texture representation and used the iterative orientation estimation from the previous section.

The second state of the art method was the classifier based orientation estimation method introduced by [2]. This method is complementary to our method, using a fundamentally different way of orientation estimation. The classifier combines HOG-based features to get an absolute orientation estimate per frame. It uses a mixture of four orientation experts, each trained on images showing pedestrians in one of four canonical orientations (front, back, left, right). The experts' classification results are used as weights in a Gaussian Mixture Model, creating a full orientation probability density function (*pdf*). A maximum likelihood estimate derived from this *pdf* is used as the final orientation estimate. The experts were trained using data kindly provided by the authors of [2]. We generated regions of interest in the 2D images based on the projection of the volume space onto the camera plane. Orientation estimates per camera are combined into a 3D orientation estimate using the camera calibration.

The following settings are used for the experiments. Textures are sampled in standard RGB colorspace. Preliminary experiments were done using both normalized RGB and C-invariant color spaces [18], but overall best results were gained using standard RGB. The voxelspace has a resolution of $2 \times 2 \times 2$ cm/voxel. Textures and shapes are sampled at a resolution of 55×55 surface points and a texture learning rate $\alpha_{\mathcal{T}} = 0.05$ is used. For the methods using *SH* based textures 9 *SH* bands are used for texture representation. For the *SH* shape representation 17 *SH* bands are used. For all methods, each frame's orientation estimate was constrained to be within 30° of the previous frame's estimate. Applying this constraint for the classifier based approach was done by limiting the range of the *pdf* when computing the maximum

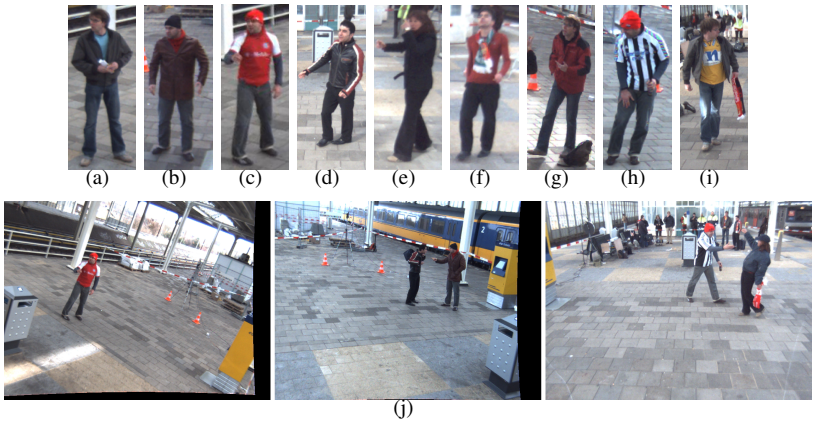


Fig. 3. (a)-(i) The 9 people used in the 12 scenarios. (j) Samples of scenarios, showing the 3 camera viewpoints and 3 scenarios.

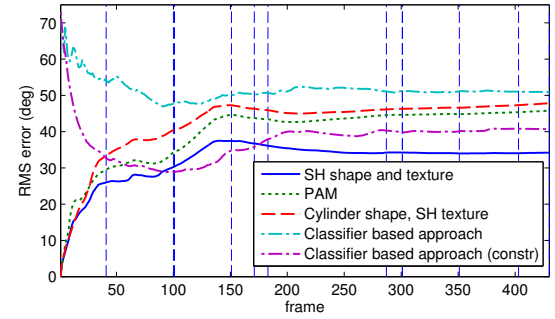
likelihood orientation. Since the classifier based approach is time independent by nature, we also tested this method without applying the 30° orientation constrained.

All methods were evaluated on about 2800 frames, distributed over 12 scenarios recorded in an open, outdoors environment with uncontrollable illumination. Three cameras with fully overlapping views are used recording at 20 Hz and at a resolution of 752×560 pixels. While some scenarios feature multiple people, orientation estimation was only performed for one person. The 9 different people involved in the scenes can be found in fig. 3(a)-(i), while fig. 3(j) shows sample frames from 3 scenarios and all camera viewpoints.

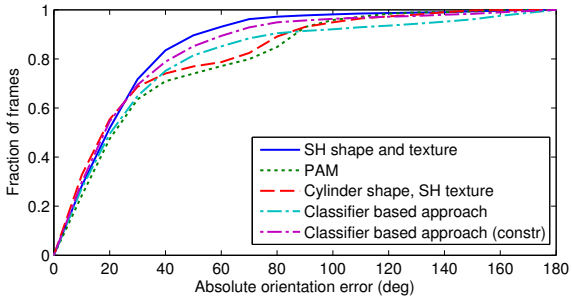
Ground-truth (GT) was created by labeling 17 distinctive locations (joint and head positions) on each person's body in each camera and using these to compute the pose of a 3D person model. The annotated torso orientation was taken to be the person body facing direction. Doing this for a regular interval of frames results in a positional accuracy of about 4 cm. We use GT foreground segmentations created using the annotated 3D pose to eliminate the effect of segmentation artifacts and to solve the problem of selecting the object of interest when multiple objects are available in the scene.

Fig. 4(a) shows the moving RMS error at time t: the RMS error over frames 1 to t of all scenarios combined. Since not all sequences have the same length, vertical bars indicate the points where scenarios end and the RMS error is taken over fewer scenarios. Please note that since the first three methods all provide relative orientation estimates w.r.t. the first frame, the error is 0 at this point. Because the classifier based approach gives an absolute orientation estimate for each frame, its error is not 0 for the first frame. Furthermore, since the moving RMS error for the first few frames has a very small sample size, the lower range of classifier based results is not representative. Comparing the classifier based approach to the other methods is best done at the later frames.

Measured over all scenarios, our method consistently outperforms both other relative orientation estimation methods. In the first 40 frames the error rises quickly because of suboptimal shape and texture models. A comparison with the method using the cylinder together with *SH* texture shows



(a) Moving RMS error evolution over time



(b) Cumulative distribution of absolute orientation error

Fig. 4. (a) Moving RMS error over all scenarios, vertical lines mark scenario end frames. (b) Cumulative absolute error distributions over all scenarios.

that our method benefits most from modeling the shape of the person. Representing the texture in low-dimensional *SH* space gives a comparable performance to PAM's full texture approach, especially when considering that the 9 *SH* bands reduce the number of features by 98%. However, when the person being tracked has a low-contrast appearance like the one shown in fig. 3(e), the *SH* texture lacks detail.

Constraining the orientation difference per frame for the classifier based method gives a significant performance improvement. While the unconstrained method is outperformed by all methods, the constrained version outperforms both PAM and the cylinder with *SH* texture. However, our method

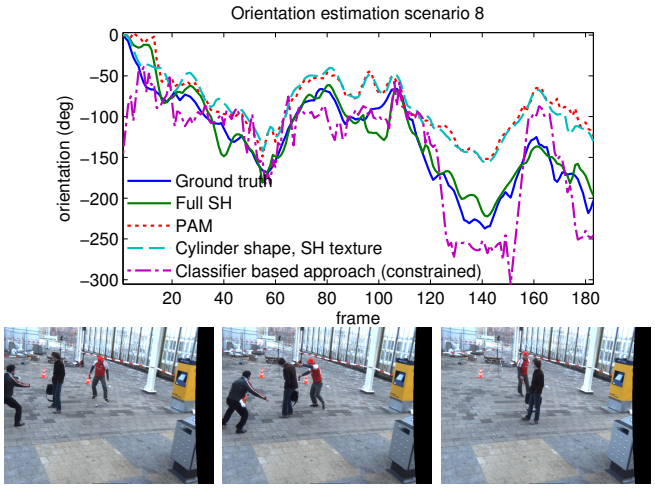


Fig. 5. (top) Example of the estimated orientation of the person from fig. 3(c) using the four methods. (bottom) Frames 48, 98 and 148 form this scenario. Notice some occlusion has occurred between frames 98 and 148.

still shows a lower RMS error compared to the constrained classifier when measured over all frames. While the quality of its orientation estimates are independent of time, the constrained classifier's performance seems to be slightly better between frame 100 and 175. This can be explained by some shorter scenarios featuring low contrast people (e.g. fig. 3(b) and (e)) whose orientation can be estimated better with the less texture-sensitive classifier based approach.

Fig. 4(b) shows each method's cumulative error distribution, obtained by binning the errors over all frames. Our method outperforms all other methods with respect to the error distribution. Most noticeable is the gap in the fraction of frames between $40°$ and $80°$ orientation error, compared to the PAM and cylinder shape method, meaning that our method has significantly less errors $> 80°$. The unconstrained classifier approach shows more errors around $180°$, matching the results from fig. 6 of [2]. These errors are caused by the ambiguity in person shape when viewed from the front or the back. The constrained classifier based method has less large errors than the other alternative methods but is still outperformed by our method.

Fig. 5 shows an example of the orientation estimation performance on a frame-to-frame basis in one scenario. In order not to clutter the graph too much, we only show the constrained classifier results here, since it gave the best overall RMS score in the previous experiments. Our method is shown to follow the GT orientation closely, while both PAM and the cylinder with *SH* texture method drift away from the GT. This drift becomes larger at frame 110, around which point in the scenario the tracked person is occluded in one of the cameras. Our method shows more robustness to this occlusion. The classifier based approach does a reasonable job following the GT, but shows a bit more erratic behavior, due to its frame-wise estimation approach.

All experiments were done on a single core of a 2.6 Ghz Intel Xeon CPU. Average processing times were: 9.5 s/fr for our method, 1 s/fr for PAM, 1.7 s/fr for the cylinder with *SH* texture and 0.7 s/fr for the classifier based approach. Volume carving and computation of the spherical shape function took about 8 s of the 9.5 s our method needs per frame, using a crude implementation. A large speed improvement is possible by using a GPU implementation. The classifier based approach was implemented in C++, while the other methods partly contain unoptimized Matlab code.

VI. Conclusion

We presented a novel approach for estimating the relative person orientation, based on a low dimensional shape and texture representation using Spherical Harmonics. Orientation estimation experiments show that this approach outperforms several alternative methods. In future work, results could be improved by using more advanced inference methods like particle filters instead of the maximum likelihood approach for selecting the position and orientation used here. Furthermore, the fusion of our relative orientation estimates with classifier based absolute orientation estimates might further improve performance.

References

[1] T. Gandhi and M. M. Trivedi, "Image based estimation of pedestrian orientation for improving path prediction," in *Proc. of the IEEE IV*, 2008, pp. 506–511.

[2] M. Enzweiler and D. M. Gavrila, "Integrated pedestrian classification and orientation estimation," in *Proc. of the IEEE CVPR*, 2010, pp. 982–989.

[3] H. Shimizu and T. Poggio, "Direction estimation of pedestrian from multiple still images," in *Proc. of the IEEE IV*, 2004, pp. 596–600.

[4] C. Chen and J. Odobez, "We are not contortionists: Coupled adaptive learning for head and body orientation estimation in surveillance," in *Proc. of the IEEE CVPR*, 2012, pp. 1544–1551.

[5] T. Gandhi and M. M. Trivedi, "Person tracking and reidentification: Introducing panoramic appearance map (PAM) for feature representation," *MVA*, vol. 18, no. 3, pp. 207–220, 2007.

[6] H. Ben Shitrit, J. Berclaz, F. Fleuret, and P. Fua, "Tracking multiple people under global appearance constraints," in *Proc. of the IEEE ICCV*, 2011, pp. 137–144.

[7] M. Liem and D. M. Gavrila, "Multi-person localization and track assignment in overlapping camera views," in *Proc. of the DAGM*, 2011, pp. 173–183.

[8] A. Mittal and L. S. Davis, "M 2 tracker: a multi-view approach to segmenting and tracking people in a cluttered scene," *IJCV*, vol. 51, no. 3, pp. 189–203, 2003.

[9] A. Balan *et al.*, "Detailed human shape and pose from images," in *Proc. of the IEEE CVPR*, 2007, pp. 1–8.

[10] J. Gall *et al.*, "Motion capture using joint skeleton tracking and surface estimation," in *Proc. of the IEEE CVPR*, 2009, pp. 1746–1753.

[11] M. Hofmann and D. M. Gavrila, "3D human model adaptation by frame selection and Shape-Texture optimization," *CVIU*, 2011.

[12] Z. Yue, W. Zhao, and R. Chellappa, "Pose-encoded spherical harmonics for face recognition and synthesis using a single image," *EURASIP*, vol. 2008, pp. 65:1–65:18, Jan. 2008.

[13] B. Bustos *et al.*, "Feature-based similarity search in 3D object databases," *ACM Computing Surveys*, vol. 37, pp. 345–387, Dec. 2005.

[14] K. Liu *et al.*, "3D Rotation-Invariant description from tensor operation on spherical HOG field," in *Proc. of the BMVC*, 2011.

[15] R. Green, "Spherical harmonic lighting: The gritty details," in *Game Developers Conference*, 2003.

[16] A. Makadia and K. Daniilidis, "Direct 3D-rotation estimation from spherical images via a generalized shift theorem," in *Proc. of the IEEE CVPR*, vol. 2, 2003, pp. II– 217–24.

[17] D. Saupe and D. V. Vranić, "3D model retrieval with spherical harmonics and moments," in *Proc. of the DAGM*, 2001, pp. 392–397.

[18] J. M. Geusebroek, R. van den Boomgaard, A. W. M. Smeulders, and H. Geerts, "Color invariance," *IEEE Trans. on PAMI*, vol. 23, no. 12, pp. 1338–1350, Dec. 2001.