An Empirical Study of Non-Rigid Surface Feature Matching of Human from 3D Video

Ashish Doshi, Jonathan Starck, Adrian Hilton

Centre for Vision, Speech and Signal Processing
Faculty of Engineering and Physical Sciences
University of Surrey
Guildford GU2 7XH
United Kingdom
email:{a.doshi,j.starck,a.hilton}@surrey.ac.uk

Abstract

This paper presents an empirical study of affine invariant feature detectors to perform matching on video sequences of people with non-rigid surface deformation. Recent advances in feature detection and wide baseline matching have focused on static scenes. Video frames of human movement capture highly non-rigid deformation such as loose hair, cloth creases, skin stretching and free flowing clothing. This study evaluates the performance of six widely used feature detectors for sparse temporal correspondence on single view and multiple view video sequences. Quantitative evaluation is performed of both the number of features detected and their temporal matching against and without ground truth correspondence. Recall-accuracy analysis of feature matching is reported for temporal correspondence on single view and multiple view sequences of people with variation in clothing and movement. This analysis identifies that existing feature detection and matching algorithms are unreliable for fast movement with common clothing.

First presented at the 5th European Conference on Visual Media Production 2008, extended and revised for JVRB

Keywords: Feature matching, sift, video sequences, recall-accuracy, qualitative analysis

1 Introduction

As identified by Lowe [Low04], object feature matching between images represents a fundamental problem in object and scene recognition in the fields of computer vision and graphics. The primary focus of this work, is to investigate the consistency of features obtained from real time capture of human motion. Therefore, identifying appropriate features, such as corners, edges and creases is important. Wide baseline correspondence detection for static and dynamic scenes has been demonstrated using multi-scale affine invariant feature detectors such as SIFT [Low04]. Recent research in temporal correspondence for non-rigid sequence of people has also employed SIFT for feature detection [ATR+08, dAST+08, SH07b]. However, only limited quantitative evaluation of feature detection and temporal matching has been performed with prior knowledge of scene structure.

This paper does not purport to investigate the properties of feature descriptors or how best to compute them. There are numerous publications that comprehensively review the virtues of such feature detectors, namely by Lowe [Low04], Mikolajczyk & Schmid [MS05], Mikolajczyk *et. al.* [MTS+05], Li & Allinson [LA08] and Tuytelaars & Mikolajczyk [TM08]. However these studies are limited to the consideration of rigid scenes. In this work, we extend the work presented in [DSH08] and evaluate the

performances of a range of detectors for multiple view video of non-rigid scenes of moving people from 3D video.

As reported in [Low04], the number of feature descriptors that can be extracted in an image is in the order of thousands. This is especially true for images with a lot of texture and fine/coarse structure. This is also true for surface correspondence of people reconstructed from multiple view video [SH07a, SMH06].

The examples shown in Figure 1 highlight the variation in number of features detected for three people with a variety of common clothing. In these images the number of features varies from around 200 keypoints for a person with uniform monochromatic clothing to over 1300 keypoints for someone with fully textured patterned clothing. Except for textured clothing, a lot of features arise from creases in clothing. There is an inherent ambiguity in matching edge features along the direction of the edge and a keypoint detector may locate the salient point at different positions along the edge across viewpoints and over time as long as the point is within view. The centre of the yellow circles plotted on each example in Figure 1 reflect the location of each feature and the size of the circle represents the scale at which the extrema of the feature has been detected. For Figure 2, additional yellow circles surrounding areas in the images depict regions that contain features.

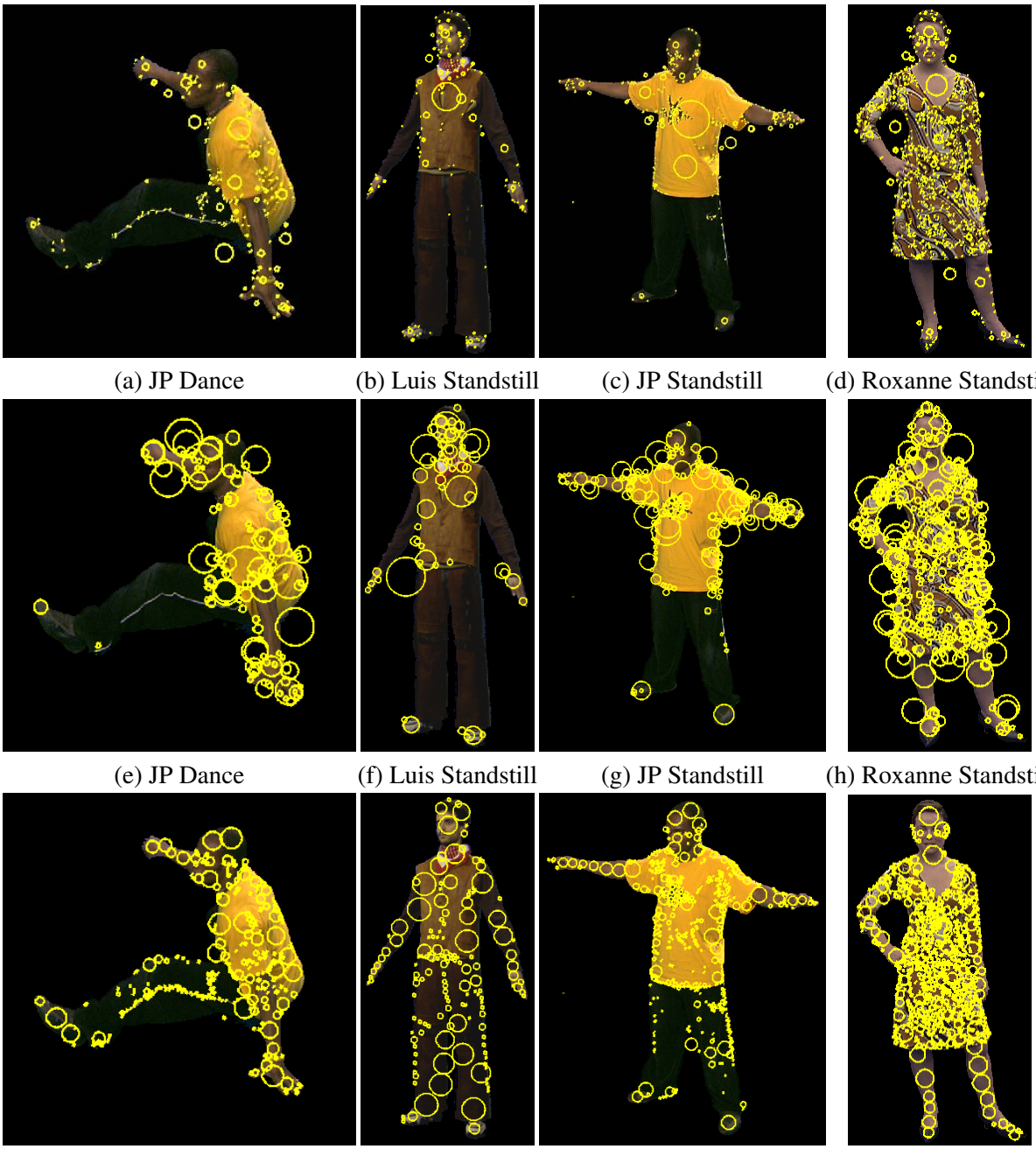
The problem remains to determine the best method to achieve an optimised 3D surface feature matching between consecutive video frames over a sequence and across multiple viewpoints. Accurate and efficient feature matching between frames is important to avoid drift in tracking the non-rigid object movement over time. It is important to note that occlusion boundaries (outline of the body or self occlusion) cannot be matched across viewpoints or across time. Previous evaluation of feature matching for static or dynamic scenes reported in [LA08, MS05, MTS$^+$05, TM08] use visual markers, to identify ground truth correspondence for comparison of feature detectors. However, for sequence of people wearing loose clothing, insertion of visual markers is difficult without affecting the natural movement. In this paper, we conduct an evaluation of feature detection and matching performance for highly non-rigid sequences of people with loose clothing. Ground truth evaluation of the feature detectors is achieved by manually marking correspondences in the captured video sequences. This allows evaluation of performance for natural clothing and movement without imposing any additional physical constraints.

This paper is organised in the following manner: Section 2 provides an overview of research conducted in the area of feature matching for wide baseline stereoscopic video sequences and in the field of graphics and visualisation. We include a brief description and parameters used (giving best possible detection and matching) of six main feature and region detectors in Section 3. This is followed by some results on unconstrained feature matching in Section 4. The proposed method to verify the consistencies of a pair of real time feature matches is described in Section 5. Also in Section 5 is the comparative analysis between feature detectors against manually labelled ground truth correspondences. Concluding remarks and possible future research directions are presented in Section 6.

2 Related Research

In recent years a significant amount of work has been conducted in shape-from-silhouettes [SMH06]. The 3D shape is reconstructed from multiple view object silhouettes segmented from the background [SH07a]. The benefit of using such a method is the possibility of reconstructing a three dimensional volume of an arbitrary moving subject. This has added realism when the subject is shown to be moving in time, as it would be in the original captured video sequence. Although the reconstructed surface is smooth and captures the non-rigid nature of the subject, three dimensional reconstruction of the local geometry of the surface representing 3D textured regions of the subject is a known problem [ATR$^+$08]. This is particularly due to lack of information of the 3D surface correspondences.

Numerous methods have been employed by researchers to obtain surface correspondences, which aid in realistic reconstruction of the subject and track their movement in the video sequence. Some of the popular methods used are SIFT [Low04], SURF [BTG06] and the Scale Saliency algorithm [KB01], although these methods are not directly used for reconstruction purposes. For example, de Aguiar *et. al.* [dAST$^+$08] used the SIFT method to extract 3D surface correspondences from multiple view video sequences. Matching correspondences between video frames are then used to constrain the Laplacian deformation of the mesh. Though no numerical values were given, they report accurate correspondence matching results. As suggested by de Aguiar *et. al.* [dAST$^+$08], ho-

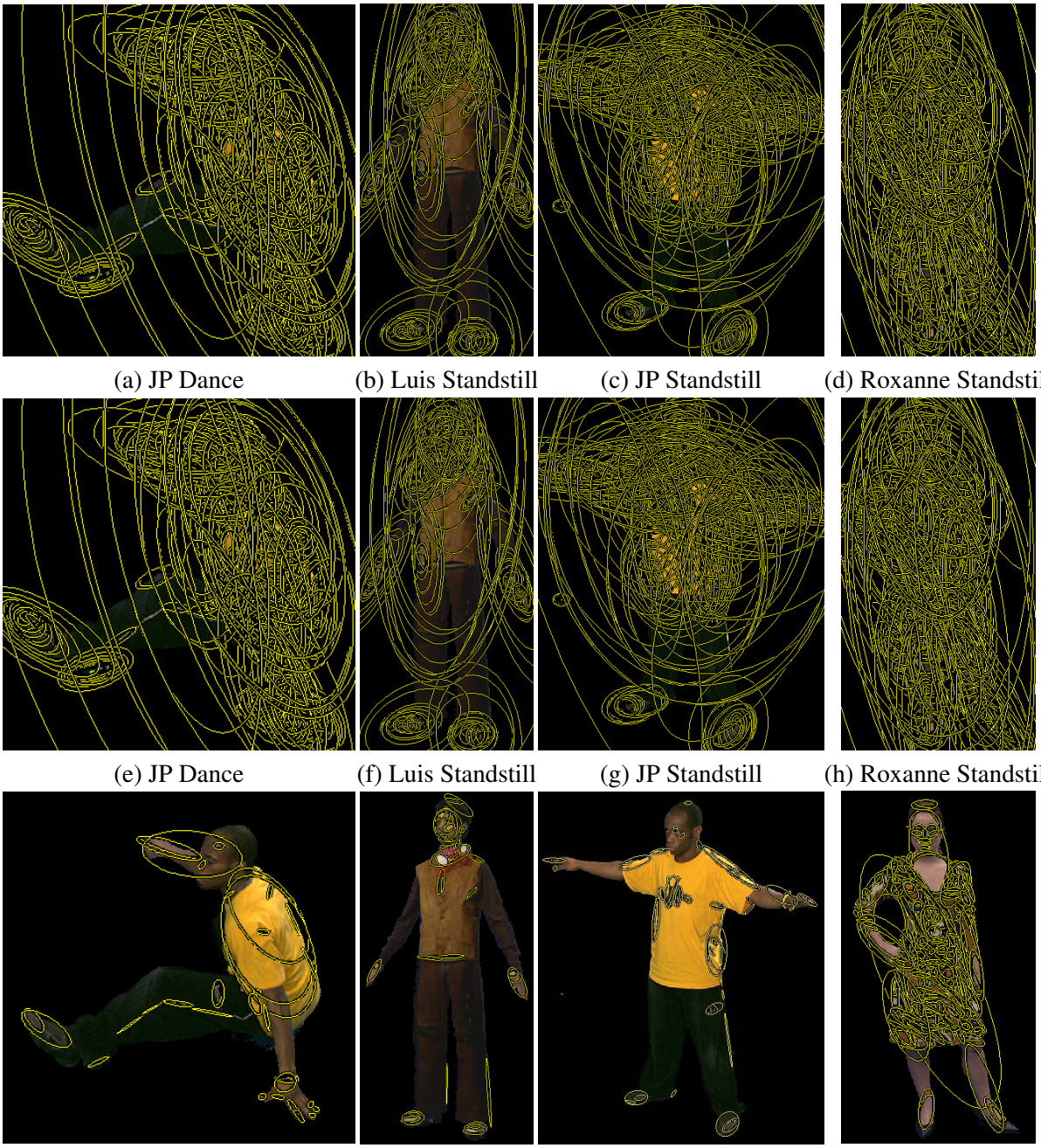


(a) JP Dance (b) Luis Standstill (c) JP Standstill (d) Roxanne Standstill

(e) JP Dance (f) Luis Standstill (g) JP Standstill (h) Roxanne Standstill

(i) JP Dance (j) Luis Standstill (k) JP Standstill (l) Roxanne Standstill

Figure 1: Identification of features using (a)-(d) SIFT; (e)-(h) SURF and (i)-(l) Scale Saliency.



(a) JP Dance (b) Luis Standstill (c) JP Standstill (d) Roxanne Standstill

(e) JP Dance (f) Luis Standstill (g) JP Standstill (h) Roxanne Standstill

(i) JP Dance (j) Luis Standstill (k) JP Standstill (l) Roxanne Standstill

Figure 2: Examples of features identified using (a)-(d) HesAff-GLOH; (e)-(h) HesAff-SIFT and (i)-(l) MSER-SIFT.

mogeneous surface regions tend to limit the SIFT method, hence their proposal of an additional silhouette based constraint. Inline with the work conducted in [dAST+08], Ahmed *et. al.* [ATR+08] used the SIFT [Low04] algorithm to extract the surface correspondences from the 3D object mesh reconstructed using a shape-from-silhouette method. The extracted feature correspondences are sparse, making it easier for them to constrain the regions surrounding each feature for region interpolation purposes. Similarly, Varanasi *et. al.* [VZBH08] used SURF to obtain an initial set of sparse correspondences. They acknowledged that in some instances, the sparse correspondences are too few for reliable mesh deformation. Hence, they proposed to use an additional coarse correspondence registration method based on normalised geodesic integral for feature detection at the limbs of the subject.

Since SIFT is the most widely used feature detector, it is natural that at some point, Lowe's algorithm and implementation would be improved depending on applications. One such variation is Ke and Sukthankar's [KS04] proposal of incorporating principal component analysis (PCA) into SIFT. Features are initially detected using SIFT. Then, for every feature detected, a 41×41 sized patch of the gradient map surrounding the keypoint is extracted. This patch is then reduced to a $2 \times 39 \times 39$ feature vector. The feature vectors are projected to PCA domain using pre-defined eigenspace transformation function. The eigenspace is made up of large number of patches learned from a large offline training set, although this only needs to be computed once. PCA is applied on the feature vectors to reduce the feature vector dimension which leads to shorter descriptor. PCA-SIFT[1] is only used for feature vector dimension reduction and feature matching. It is dependent on Lowe's SIFT algorithm to detect features and construct the initial 128-bit descriptor.

More evidence of surface correspondences being used for wide baseline non-rigid surface matching from 3D video sequences is presented in [ASK+04, SH07b]. In [ASK+04] an unsupervised learning algorithm is introduced that searches for matching correspondences between frames based on the geodesic distance over the surface mesh. The limitation of this approach is that a significant number of scans are needed in the learning phase to robustly reconstruct meshes and generate matching correspondences. Secondly, as reported in [ASK+04], the method is sensitive to outliers. Hence, it is important that decluttering of the noise take place in the pre-processing stage. In contrast to [ASK+04], Starck and Hilton [SH07b] proposed using the maximum likelihood of the Markov random field for matching and tracking. This results in accurate correspondence matching between meshes with large non-rigid shape differences but has a relatively high computational cost.

Also of relevance is the work conducted recently by Tola *et. al.* [TLF08] and Ozuysal *et. al.* [OFL07]. In [TLF08], the authors proposed the DAISY feature descriptor which is similar to that of SIFT [Low04] and GLOH [MS05] but faster and more efficient. Instead of using weighted sums [Low04, MS05, BTG06] in the descriptor, Tola *et. al.* used convolutions of the original image with oriented derivatives of Gaussian kernels resulting in quicker construction of the descriptor, although not significantly different to SIFT and GLOH. It is worth noting that DAISY uses circular bins compared to rectangular bins as in [Low04, MS05, BTG06] which makes it more tolerant to orientated features. Similar to [ASK+04, SH07b], Ozuysal *et. al.* presented a learning based technique to detect features in [OFL07]. They proposed using a Naive Bayesian classification approach in a non-hierarchical structure referred to as ferns to classify patches of features. Although the methods reported in [TLF08] and [OFL07] seem to either outperform or have similar performance to SIFT in static scenes, these methods are relatively new and have as yet not be applied for feature detection in multiple view video sequences of people.

In recent years, lot of focus has been placed on detecting features in the 2D domain for matching and tracking of 3D surfaces. However, it is also worth mentioning that methods do exist to perform detection and matching in three dimensional space. Starck *et. al.* [SMH05] proposed an animation control algorithm based on motion graph and spherical matching method to optimise mesh blending. Their motion blending approach incorporates a coarse-to-fine optimisation algorithm which is dependent on multi-view surface correspondences. On a similar note, Hilaga *et. al.* [HSKK01] used normalised geodesic integrals to segment the surface geometry into patches. Subsequently, Reeb graph is used to find the optimal topological matches between surfaces. More recently, Zaharescu *et. al.* [ZBVH09] presented MeshDOG and MeshHOG for surface feature detction. MeshDOG is a 3D surface feature detector that performs difference

[1] http://www.cs.cmu.edu/ yke/pcasift/

of Gaussians on the surface geometry instead of texture images as other standard detectors do. Simply put, their detector is based on finding the appropriate surface correspondences using the Hessian. MeshHOG is the descriptor counterpart of the surface correspondence. Although, they yielded good results, optimum matching strategy of the surface correspondences is still based on a user-defined threshold.

Other related work that would be of some interest in the near future is using dense correspondences for 3D surface tracking. Brox *et. al.* [BBM09] reported using total variational optical flow method to track dense correspondences in a patch from monocular video sequences. In contrast, Vedula *et. al.* [VBK02] used optical flow in 3D to track dense surface correspondences in time. Though not widely used, there is plenty of scope to use 3D optical flow for non-rigid deformable surface matching and tracking.

3 Detectors

This study is conducted on video sequences captured in a multiple camera studio. The studio comprises of 8 cameras equally spaced in a circle of radius 4m at a height of 2m oriented towards the centre giving a 45^o angle between views. Video sequences are 1920×1080 uncompressed 25Hz HD-SDI progressive scan format. Performances were captured under uniform illumination with blue screen chroma-key backdrops to facilitate foreground segmentation. A number of feature detectors have been evaluated to verify their suitability for feature matching on sequences of people[2].

3.1 Scale Invariant Feature Transform (SIFT)

SIFT [Low04] is one of the most widely used technique in human motion capture for identifying affine invariant features. The SIFT algorithm makes use of the Hessian of initial features (obtained from difference of Gaussians between scales) to robustly detect features that are invariant extrema. SIFT features have recently been used for frame-to-frame tracking of non-rigid motion to obtain temporally consistent multiple view reconstruction of subjects [AdAT$^+$05, dATSS07, SH07b].

The examples shown in Figure 1(a)-(d) are results obtained using the standard SIFT [Low04] method with default parameters on images of three people wearing a variety of clothing. The parametric values for SIFT are: standard deviation of the Gaussian function, $\sigma = 1.6$; scaling factor, $\kappa = \sqrt{2}$; number of octaves per scale, $oc = 3$; extrema points to be discarded, $|D(\hat{\mathbf{x}})| < 0.03$; ratio of principal curvatures, $r = 10$. From Figure 1(a)-(d), it is observed that SIFT is able to distinguish clearly structures with textural information. It is unable to extract good features from colour gradients or from homogenous coloured clothing or appearance. Furthermore, there is some degree of success when identifying features that define creases and folds.

3.2 Speeded Up Robust Features (SURF)

The SURF feature detector *et. al.* [BTG06] is a relatively new method for computationally efficient feature detection. In contrast to using the difference of Gaussians (DOG) by Lowe [Low04], the SURF method is based on using the determinant of the Hessian matrix to search for locations and scales of unique features. Furthermore, the descriptor of these features are obtained by approximation of Gaussian based box filters on the integral image, rather than the input image, which can be evaluated more efficiently. The parametric values for SURF are: number of octaves per scale, $oc = 3$; length of descriptor vector, $e = 128 - bit$.

Application of SURF feature detection on the same set of example images is shown in Figure 1(e)-(h). Although the number of features obtained is less compared to SIFT, it can be observed that the features are concentrated around edges of the clothing and distinct object structures.

In terms of speed of the detector, SURF is clearly quicker compared to SIFT, as can be observed in Table 1 and Table 2. SURF is quicker than SIFT because it detects less features and thus require less computation. In this comparison neither of the algorithms implementations tested have been optimised. Default parameters for the detectors were used in all cases as this has been found to give good results.

3.3 Scale Saliency (SS)

The scale saliency (SS) feature detector [KB01] is increasingly being used to detect salient features. SS identifies features that are unique and prominent in an object [HL03]. The detector uses an entropy based method that measures the predictability or unpredictability of local intensities for a feature. If the

[2] http://kahlan.pdf.surrey.ac.uk/cvssp3d/

feature exhibits high entropy, then it is deemed salient. The parametric values for SS are: number of octaves per scale, $oc = 3$; anti-aliasing, $AA = 0$; number of bins, $nbins = 8$; standard deviation of the Gaussian function, $\sigma = 1.6$; threshold on saliency values, $wt = 0.5$; threshold on inter-scale saliency, $yt = 0$.

Figure 1(i)-(l) shows application of the SS detector on example images of people. In the study performed, the SS method detects a far larger number of features compared to SURF and is comparable to SIFT. The method has detected a lot of regions with corners and edges, especially on the boundary of the object. Scale saliency has also determined that regions with illumination that change from one patch to another can be considered as features. It is important to note that initially, SS detects features in the region of tens to hundreds of thousands. However, a greedy clustering mechanism is integrated to reject outliers and keep regions with stable high entropies.

3.4 Hessian Affine - GLOH (HesAff-GLOH)

In this study, 3 top performing region detectors are used for additional comparisons against the state-of-the-art feature detectors. These region detectors use either GLOH or SIFT to compute feature descriptors. HesAff-GLOH is a region detector that uses an additional GLOH (gradient location and orientation histogram) capability to detect features within a region. The Hessian affine region detector was proposed in [MTS$^+$05]. It is based on using the second derivatives of the image to detect blobs and ridges. The affine neighbourhood of the region is determined by the affine adaptation process as suggested in [MTS$^+$05]. The GLOH capability is a type of feature descriptor method that has been developed by the authors in [MS05]. It is an extension of the SIFT descriptor method, but more robust.

Figure 2(a)-(d) shows examples of features detected using the region detector. As it can be observed, each region that contains unique features are also considered features themselves. The HesAff-GLOH detector finds regions with concentrated features, yielding identifiable regions such as shoes, wrists, hands, head and other noticeable regions.

3.5 Hessian Affine - SIFT (HesAff-SIFT)

The HesAff-SIFT method as proposed in [MS05] is quite similar to HesAff-GLOH in that it uses the same Hessian affine region detector. However, instead of using GLOH for computing feature descriptors, HesAff-SIFT uses SIFT. Although the examples shown in Figure 2(e)-(h) are similar to that of HesAff-GLOH, there are some minor differences between the two detectors. As presented in Table 1, HesAff-SIFT obtains fewer detection of features compared to HesAff-GLOH despite taking similar amount of computation time.

3.6 Maximally Stable Extremal Regions - SIFT (MSER-SIFT)

The MSER region detector [MCMP02] searches for regions where all the pixels within the region boundary has either lower or higher intensity than the pixels on the boundary of the region. The connected binarisation of the region detector is maximally stable using optimised thresholding. SIFT is used in tandem to compute for feature descriptors. As it can be observed in Figure 2(i)-(l), the MSER-SIFT detector performs rather well at detecting regions with unique features, *i.e.* texture patches with change in intensity. This is especially good for identifying body parts such as feet, hands and face.

3.7 Speed Evaluation

This study has been conducted on an Intel (R) Quad Core (TM) Q6600 2.4GHz workstation running on 3Gb of memory and Fedora Core 6. The external binaries for the detectors used were executed within MATLAB 2007b for a fair comparative analysis.

The data presented in Tables 1, 2 and 3 highlight the difference in performance for six different feature detectors, SIFT[1], SURF[2] and SS[3], HesAff-GLOH[4], HesAff-SIFT[4] and MSER-SIFT[4]. The speed comparisons are made for single frames (Table 1), single view video sequences (Table 2) and for multiple view video sequences (Table 3). The number of surface features detected is represented by $\mathcal{F}$, with the minimum number of features in a frame being $\mathcal{F}_{min}$, maximum number of features $\mathcal{F}_{max}$, averaged detected in a sequence $\mathcal{F}_{ave}$ and the amount of time needed to complete the computations is t seconds. The numbers in $(\cdot)$ represent the number of frames processed in the sequence.

The overall view of the numerical results in Table 1-3 show that HesAff-GLOH consistently outperforms

[1] http://www.cs.ubc.ca/~lowe/keypoints/
[2] http://www.vision.ee.ethz.ch/~surf/download.html
[3] http://www.robots.ox.ac.uk/~timork/salscale.html
[4] http://www.robots.ox.ac.uk/~vgg/research/affine/

	JP Dance		Luis		JP Standstill		Roxanne	
	$\mathcal{F}$	t (s)	$\mathcal{F}$	t (s)	$\mathcal{F}$	t (s)	$\mathcal{F}$	t (s)
SIFT	306	7.00	177	2.07	389	6.95	1232	7.26
SURF	123	1.24	62	0.56	213	1.25	420	1.42
SS	329	1.99	138	0.44	478	1.55	918	1.80
HesAff-GLOH	**738**	2.64	**457**	0.80	**1095**	2.79	**2618**	3.27
HesAff-SIFT	551	2.82	386	0.80	872	2.77	2110	3.30
MSER-SIFT	36	**0.78**	42	**0.38**	63	**0.60**	296	**0.60**

Table 1: Processing time and number of features for single frames.

	JP Free (500 frames)				Roxanne Twirl (46 frames)			
	$\mathcal{F}_{min}$	$\mathcal{F}_{max}$	$\mathcal{F}_{ave}$	t (s)	$\mathcal{F}_{min}$	$\mathcal{F}_{max}$	$\mathcal{F}_{ave}$	t (s)
SIFT	147	629	351	3470	919	1568	1294	335
SURF	43	283	141	600	277	500	405	68
SS	164	593	373	829	504	992	828	85
HesAff-GLOH	**328**	**1611**	**783**	1300	**1721**	**3158**	**2567**	145
HesAff-SIFT	254	1217	590	1354	1335	2475	2031	156
MSER-SIFT	24	116	67	**296**	180	332	273	**28**

Table 2: Processing time and number of features for single view video sequence.

	JP Free (4000 frames)				Roxanne Twirl (688 frames)			
	$\mathcal{F}_{min}$	$\mathcal{F}_{max}$	$\mathcal{F}_{ave}$	t (s)	$\mathcal{F}_{min}$	$\mathcal{F}_{max}$	$\mathcal{F}_{ave}$	t (s)
SIFT	103	638	331	27594	813	1910	1292	5038
SURF	43	283	143	4940	232	603	391	1010
SS	95	617	335	6592	434	1157	785	1239
HesAff-GLOH	**328**	**1611**	**819**	10291	**1518**	**3663**	**2425**	2087
HesAff-SIFT	254	1217	614	10514	1186	2925	1923	2257
MSER-SIFT	9	117	54	**2371**	141	380	258	**414**

Table 3: Processing time and number of features for multiple view video sequence (8 cameras).

all the other detectors in the test in terms of number of features detected, while MSER-SIFT has proved to be the quickest in the test. Overall, both HesAff-GLOH and HesAff-SIFT detects more features than SIFT (with SIFT shown to be better than SURF and SS), suggesting that the Hessian affine based region detector are better at locating features and also almost doubly fast with GLOH outperforming SIFT with its robustness [MS05]. Though MSER-SIFT is fast, with the lowest number of features detected, the detector is in fact the worst performer of all the detectors. This fact is understandable since MSER identify regions with noticeable intensity differences. Hence, where the subject's clothing is rather plain, MSER finds low number of features and when the subject is more textured, then the number of features detected increases. Please note that both SS[3] and MSER-SIFT[4] implementations do not compute the corresponding descriptor for each keypoint. Therefore, if the computation for 128 element descriptors are added to the speed tests in Table 1-Table 3, then SS and MSER-SIFT will be comparably slower in its overall processing.

By referring back to the examples presented in Figure 1(a)-(d), it can be observed that there are many features that are incorrectly detected within the frames. Some are embedded in the background. These features could be eliminated by segmentation of the performer from the background. Features with overlapping between foreground and background step edge may also be erroneous and can be eliminated by prior

foreground segmentation. This would remove any errors in feature matching at the boundary.

4 Feature Matching

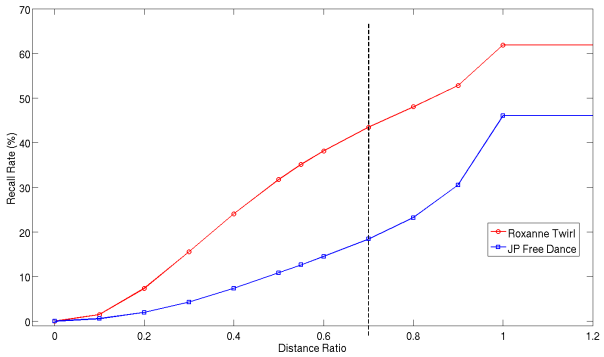


Figure 3: Estimation of β for video sequences of people.

In this section we present qualitative results of applying SIFT to sequences of people both over time for a single viewpoint and between camera views. Quantitative evaluation of matching performance is presented in Section 5. The non-rigid surface feature matching is conducted using the nearest neighbour search (NNS) method [RKD08] as follows:

$$D_e(i,j) = \| \mathcal{F}_i(t) - \mathcal{F}_j(t+1) \| \quad \forall i \in \mathcal{F}(t), j \in \mathcal{F}(t+1) \quad (1)$$

where $D_e(i,j)$ is the Euclidean distance list between two feature sets, $\mathcal{F}_i(t)$ is the feature set at time t and $\mathcal{F}_j(t+1)$ is the feature set at time $t+1$. Equation (1) states that the distance between two feature keypoints is computed by taking the Euclidean norm $\| \cdot \|$, between descriptors of the keypoints. The features would be considered matched if the distance between them is shorter than β times the distance of the second nearest neighbour [MS05]. This is given as

$$D_e(i,j) \leq \beta D_e(i,j+1) \quad \forall i \in \mathcal{F}(t), j \in \mathcal{F}(t+1) \quad (2)$$

where β is an arbitrary distance ratio. For the video sequences in this study, β is estimated to be 0.7. This value is chosen for high recall rate before breaking down. The average effect of the recall rates by varying the distance ratio is shown in Figure 3 for all frames in the Roxanne Twirl and JP Free video sequences across eight cameras. The figure shows that starting from 0.8, the recall rate starts to stabilise and does not change for any values larger than 1. Although the recall rate at this point is at its highest for the relevant sequences, the number of false matches is also high. Hence, a conservative value of 0.7 is chosen for consistent matching between video frames. This gives a reasonable balance of correctly identified matches to false matches. The number of false matches is significantly reduced, providing a clearer picture of the features that can be used for tracking purposes.

It is important to note that prior to feature matching, the number of features from the initial detection are reduced by removing features located in the background and those that are located within the error boundary (due to segmentation errors) upto 3 pixel distance.

5 Evaluation

Evaluating the performance of feature/region detectors for dynamic scenes with motion blur is difficult due to the absence of ground-truth correspondences.

How best can we evaluate the detection rate or accuracy of any method chosen for a sequence?

In this work, a measure of left-right matching consistency is proposed to evaluate matching performance for real sequences. This measure is evaluated against ground truth hand-labelled correspondences. Similar to equation (1) in the left-to-right direction, the right-to-left direction is given by

$$D_e(j,i) = \| \mathcal{F}_j(t+1) - \mathcal{F}_i(t) \| \quad \forall i \in \mathcal{F}(t), j \in \mathcal{F}(t+1) \quad (3)$$

Hence, the LR-consistency check is determined by

$$\| D_e(i,j) - D_e(j,i) \| \leq \tau \quad \forall i \in \mathcal{F}(t), j \in \mathcal{F}(t+1) \quad (4)$$

which states that two feature keypoints are considered as a *consistent match* if the Euclidean norm between the Euclidean matches, $D_e(i,j)$ (1) and $D_e(j,i)$ (3) is within an acceptable tolerance, $\tau = 0.03$.

Figure 4 shows example matching results using LR-consistency for JP Free Dance, Roxanne Twirl and Roxanne Stagger video frames. The benefit of LR-consistency is that despite using a reasonably high value for distance ratio, the consistency checks between descriptors allow for higher recall rate than it would be if a low distance ratio value were to be used.

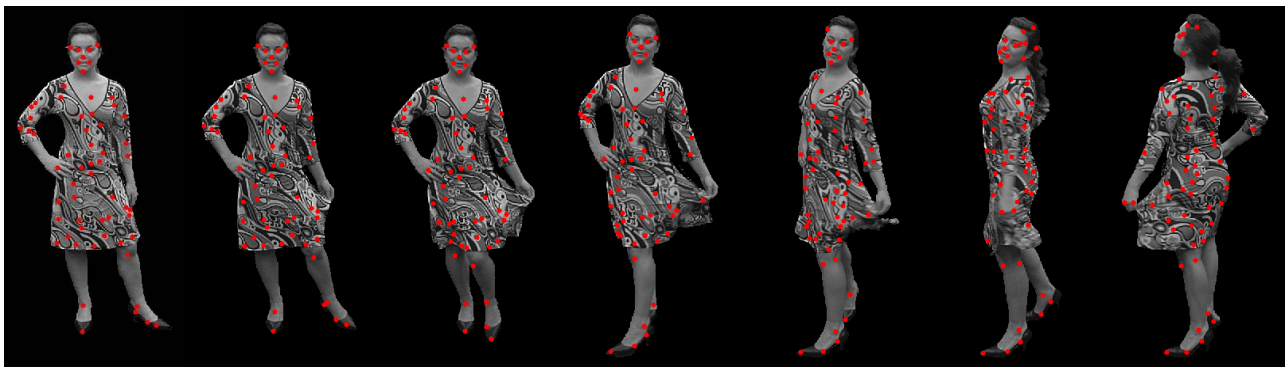


Figure 5: Video frames of Roxanne Twirl sequence with hand labelled markers in red.

This means that a higher percentage of features can be used to better capture deformations during subject movements.

However, in order to reliably compare between various feature detectors, manually labelled markers of a test sequence is required. A short sequence of about 46 video frames of Roxanne twirling in a textured dress is labelled, skipping three frames between marked frames. With reference to Figure 5, each video frame is labelled with 60 markers. For this particular sequence, 12 video frames have been labelled with 60 features per frame giving a total of 720 markers for reference. Since the actress is twirling and the scene captured is dynamic, therefore markers are discarded and added from frame to frame, as and when appropriate.

For marker based analysis, we follow the lead of the work in [Faw06] for understanding the receiver operating characteristic (ROC) measure and the recall-accuracy measure as reported in [KZ07]. The following is the confusion matrix for the recall-accuracy analysis for this study.

Classifiers	Features	
	Positives (P)	Negatives (N)
True (T)	True Positives (TP)	False Positives (FP)
False (F)	False Negatives (FN)	True Negatives (TN)

Table 4: Recall-accuracy confusion matrix.

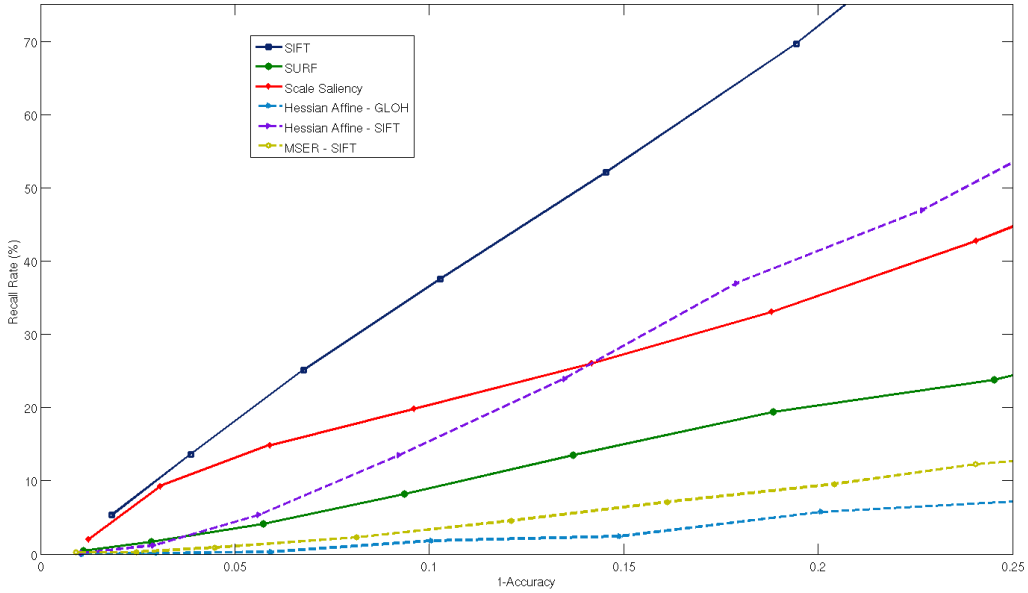
The interpretation for the matrix is as follows: Positives (P) - features that are within pixel distance, pd (varied to produce recall-accuracy graph) of the ground truth markers which are classified as positive; Negatives (N) - features that do not meet the positive criteria from the query; True Positives (TP) - features that are labelled positive and has successful positive feature match, thereby classified as positive; False Negatives (FN) - features that are labelled positive, but no corresponding matches, thereby classified as negative; False Positives (FP) - features that labelled negative, but has positive matches, hence classified as positive and True Negatives (TN) - features that are labelled negative and have correctly not matched to ground truth markers, hence classified as negative. For the recall-accuracy analysis, the measures to be computed are

$$Recall = \frac{\#correct\_matches(TP)}{\#positive\_correspondences(P)} \quad (5)$$

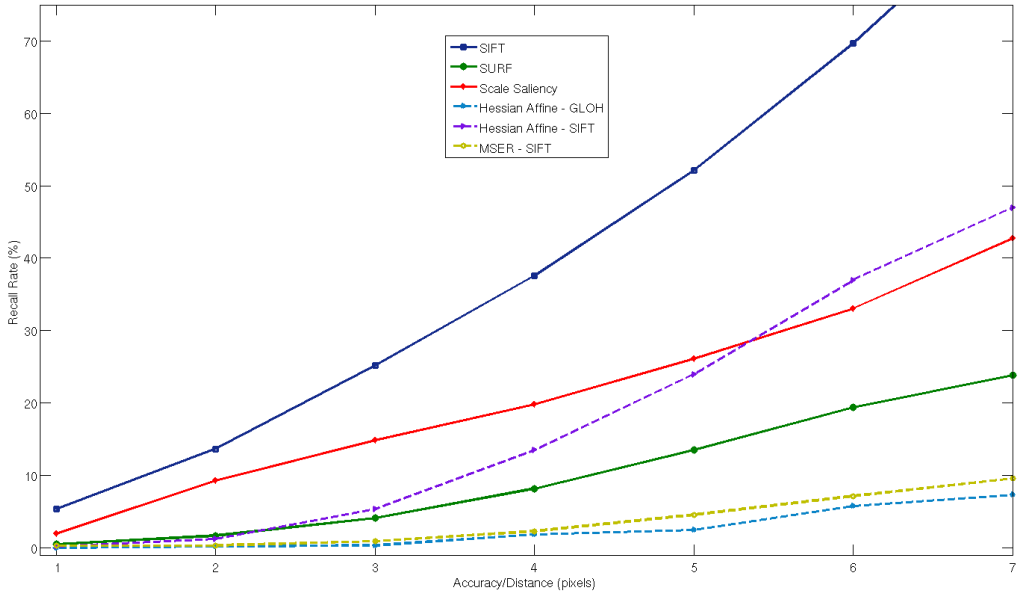
$$FalsePositiveRate = \frac{\#false\_matches(FP)}{\#negative\_correspondences(N)} \quad (6)$$

$$Accuracy = \frac{TP + TN}{P + N} \quad (7)$$

Figure 6 shows the recall-accuracy graphs of single view Roxanne Twirl video sequence (Figure 5). This is a comparison between feature detectors (SIFT, SURF and SS) and region detectors (HesAff-GLOH, HesAff-SIFT and MSER-SIFT) evaluating the performance of the respective recall rates with varying distances between ground truth feature and detected feature keypoints averaged across the whole sequence. From Figure 6(a), we can conclude that for about 90% accuracy, highest recall rate achieved is approximately 36% by SIFT despite only obtaining the third highest number of features detected in this sequence (see Table 2). This occurs when the maximum radial distance

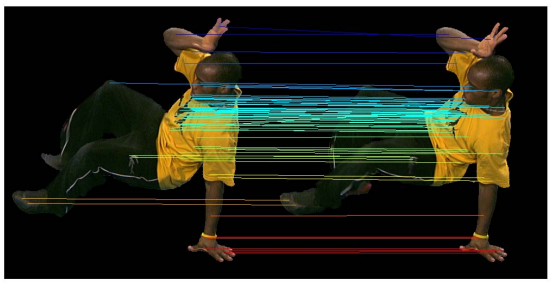


(a) Matching between every 4th frames (6fps): Recall-Accuracy

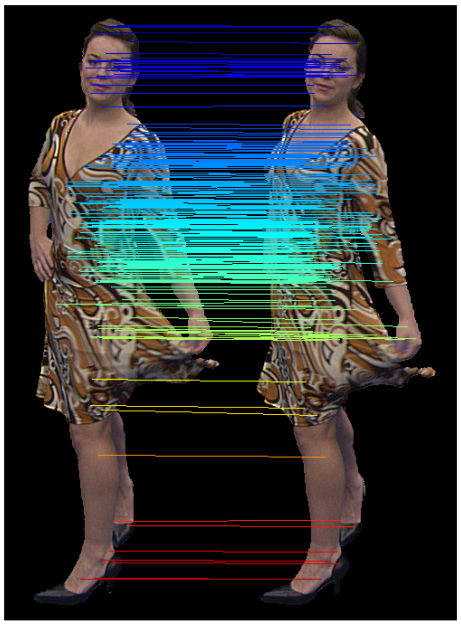


(b) Matching between every 4th frames (6fps): Recall-Distance

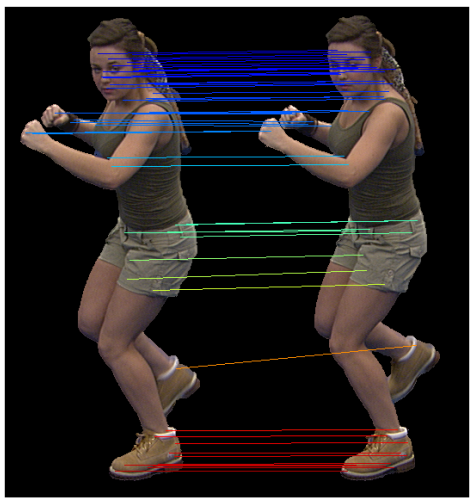
Figure 6: Single view recall-accuracy graphs for all 6 detectors on Roxanne Twirl video sequence.



(a) JP Free Dance (68 matches)



(b) Roxanne Twirl (239 matches)



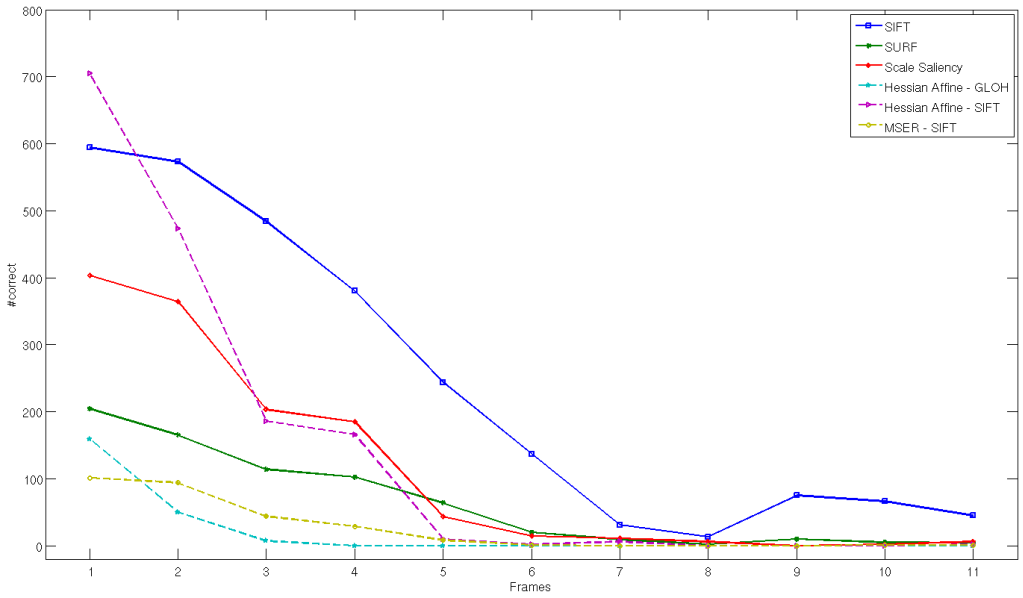
(c) Roxanne Gamecharacter (55 matches)

Figure 4: Examples of optimised feature matching results on single view video sequences in 2D.

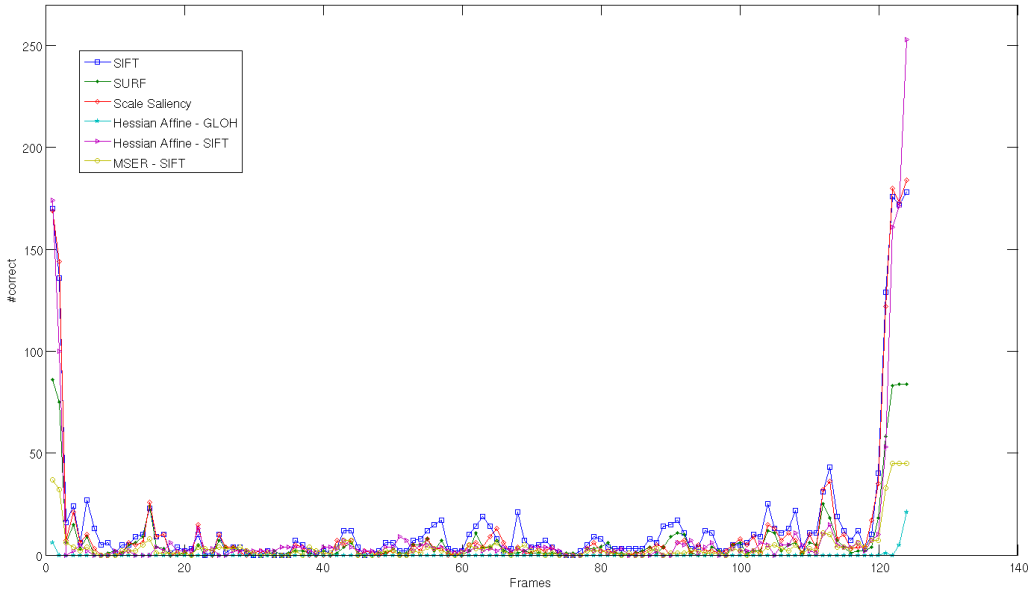
between ground truth and features detected by SIFT is 4 pixels as shown in Figure 6(b). Although initially SS is the second best performer in feature matching, HesAff-SIFT starts to obtain better matching performance once the distance radii is increased, *i.e.* the ground truth marker is within the Hessian affine detected region, hence matching could be performed. With MSER-SIFT, it is assumed that due to the low number of detected features and poor distribution in the image, this results in low recall rate. However, it is probable that after optimising the feature selection process and poor distribution of the remaining features means that HesAff-GLOH is worst in the feature matching study even with the highest initial feature detection. The cause of the low recall rate is the number of features that are within the radial distance of the ground truth markers. Ground truth markers have been placed where there are visible and distinct features to the human eye. The markers have been distributed as evenly as possible.

The reason SIFT [Low04] performed well is mainly due to the number of features that have been detected all over the subject, *i.e.* evenly distributed on the subject. It is interesting to note that scale saliency (SS) [KB01] performed poorly compared to SIFT, but better than SURF despite averaging around 830 features per frame. One factor is that the feature matching of the sequences is performed between every 4th video frame (subsampling at approximately 6fps), reducing the number of possible matches significantly. Another factor is that the subject is twirling at a higher angular speed compared to the speed of the video capture, causing blurred video frames, resulting in low number of features detected. A similar conclusion can be attributed to the performance of the SURF [BTG06] detector.

If we were to consider that ground truth markers were not available for comparative analysis purposes, then results in Figure 7 are obtained for Roxanne Twirl and JP Free Dance sequences. For Figure 7(a), it can be clearly observed that SIFT consistently achieves the highest correct matches in the subsampled sequence compared to all the other detectors. When the subject is standing still, HesAff-SIFT obtains higher number of correct matches due to its significantly high number of features detected. More features are detected by SS than SURF, although there is hardly any difference between the two when the subject is in motion. Across this short video sequence, SIFT correctly matches approximately 2650 features with 90% accu-



(a) Roxanne Twirl, matching between every 4th frames (6fps)



(b) JP Free Dance, matching between every 4th frames (6fps)

Figure 7: Single view matching results without ground truth for all 6 detectors.

racy. In comparison to SIFT, HesAff-SIFT matches approximately 1550 features, SURF matches approximately 700 features and SS matches approximately 1300 features. For subsampled JP Free Dance sequence in Figure 7(b), SIFT matches approximately 1900 features compared to SURF (900), SS (1550), HesAff-GLOH (30), HesAff-SIFT (1270) and MSER-SIFT (500), suggesting that SIFT is by far the most consistent feature detector and feature matching for people based video sequences. It should be noted that although HesAff-SIFT and SS matches more features at the beginning and end of the sequence (due to the fact that the subject is inherently motionless in the corresponding frames), SIFT is consistently better during subject movement.

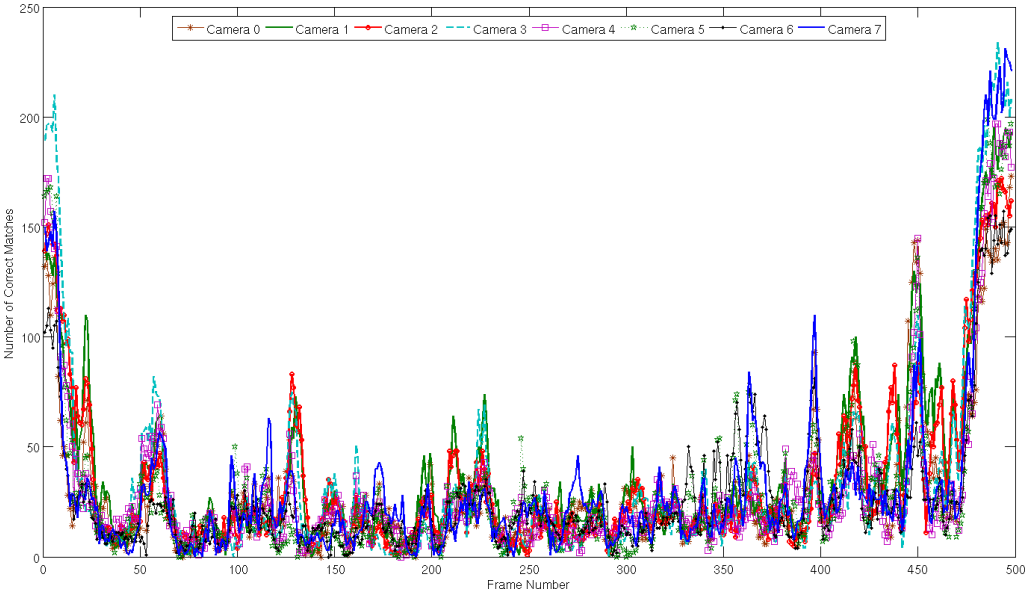
One way to evaluate the matching performance of methods for multiple viewpoint sequences is number of feature matches, $D_e^C(i,j,t)$ for the whole sequence, from frame to frame for each camera view. It would then be possible to determine the global movement of the subject. As is shown in Figure 8 for the JP Free Dance sequence, a high number of matches represent either stationary or slow movement and troughs represent fast movement (either translational or rotational) from the subject where there are significant changes in feature visibility between frames, *i.e.* motion blur. Despite downsampling the frame rate of the sequence, the overall tracking of the movements in Figure 8(b) is identical to the tracking in Figure 8(a). Similar results to Figure 8 were obtained in Roxanne Twirl sequence, thereby entertaining the notion that slow and fast movements of the subject can be tracked across the whole sequence. However, the results shown in Figure 8 are those obtained using SIFT (with SIFT being best at feature matching/tracking out of the methods evaluated).

Figure 9 presents the multiple view feature matching performances of SIFT [Low04] (selected for best results) for three video sequences subsampled at approximately 6 frames per second and two others without subsampling, *i.e.* 25 frames per second. The video sequences used are of Roxanne Twirl (6fps), JP Free Dance (6fps), Roxanne Gamecharacter Stagger (6fps), Roxanne Walk (25fps) and Roxanne Gamecharacter Jog (25fps). For each video frame, the total number of correct feature matches reported are the sum of correct matches across 8 high definition cameras. Firstly, both Roxanne Twirl and Walk sequences yield a high number of correct matches. This result is not entirely unexpected, considering that the subject is wearing highly patterned clothing, with repeated patterns. The Twirl sequence reports the highest number of correct matches because the subject is moving rotationally about the same position, while in Walk sequence, the subject is walking in a constant speed translationally. We observed that at the extreme ends of the plot the total number of matches are significantly higher due to the subject standing still. The plot also shows that as the subject is in high velocity motion, as in the case here, the number of matches (even with multiple view points) drops off. However, what is unexpected is the approximate total of 41000 correct feature matches, yielding an average of 254 correct feature matches per camera and per frame. These values are significantly higher than typical values reported in [ATR+08].

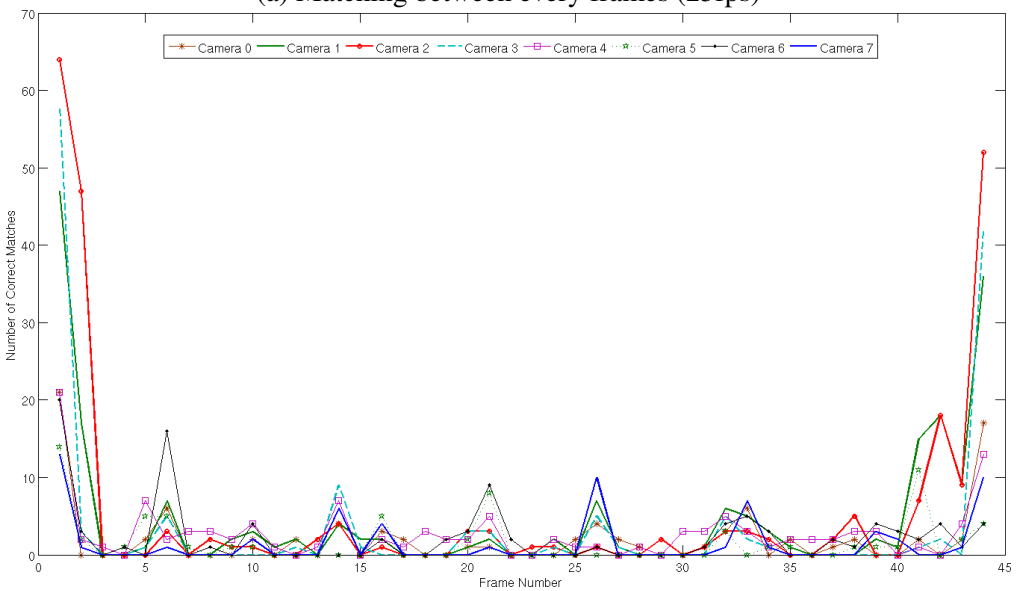
The multiple view feature matching results shown in Figure 9 highlight the differences between no motion, constant motion and fast motion. The Walk sequence is a very good example of constant motion, since its corresponding plot is relatively flat. For the other three sequences, they yield a low number of feature matches due to the subject wearing typical male attire, *i.e.* plain or single lined trackbottom and a simple pattern on an otherwise plain t-shirt; or when the female subject is wearing neutral coloured clothing. In the Stagger sequence, the female subject staggers briefly twice, resulting in troughs in its corresponding plot. Since the subject is moving faster in the Jog sequence compared to Walk sequence, a higher amount of motion blur is introduced thereby reducing the number of correct matches. The Jog sequence is also characterised as constant motion since its corresponding plot is relatively flat.

Lastly, the JP Free Dance sequence is such that the subject is initially standing still with outstretched arms and then starts to perform a free dance routine and ends back in the starting position, all in 20 seconds resulting in 500 video frames. This is then subsampled to obtain a 124 frame sequence. For comparable brevity against other motion plots, the dance sequence is further subsampled by a factor of three. We observe that this motion plot closely resembles that of Twirl plot which has both standstill and fast motion phases.

Figures 10(a)-(c) show the single view matching results on a 3D surface of a mesh. These results correspond to the 2D matching results shown in Figure 4. For each of the figures in Figure 10, there exists a starting mesh and the target mesh overlapping the former. The coloured lines in between the meshes are links of the corresponding matched features between the two



(a) Matching between every frames (25fps)



(b) Matching between every 10th frames (2.5fps)

Figure 8: Multiple view JP Free Dance long sequence tracking using SIFT feature matching.

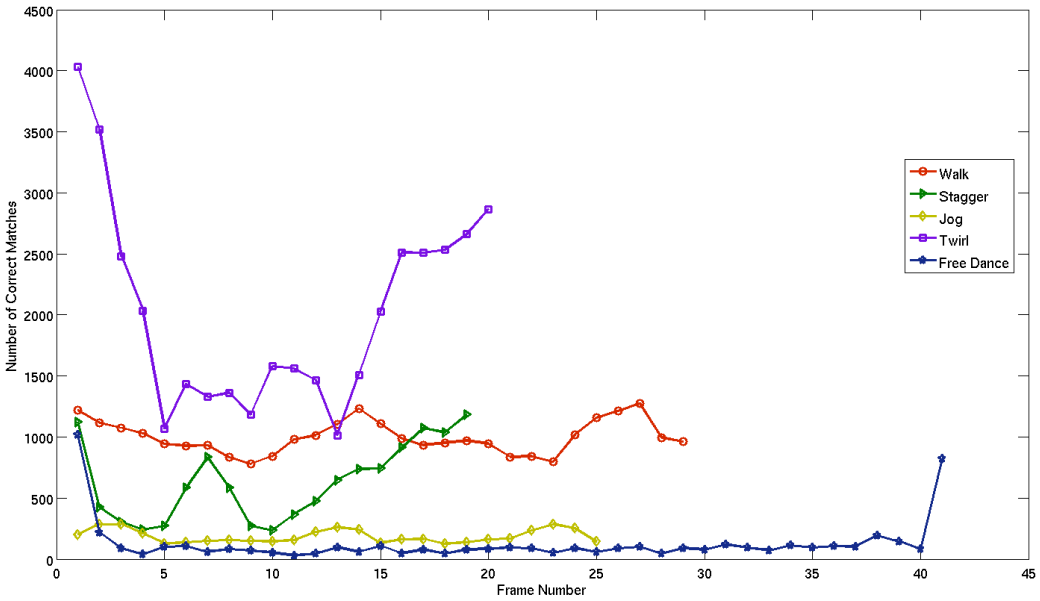
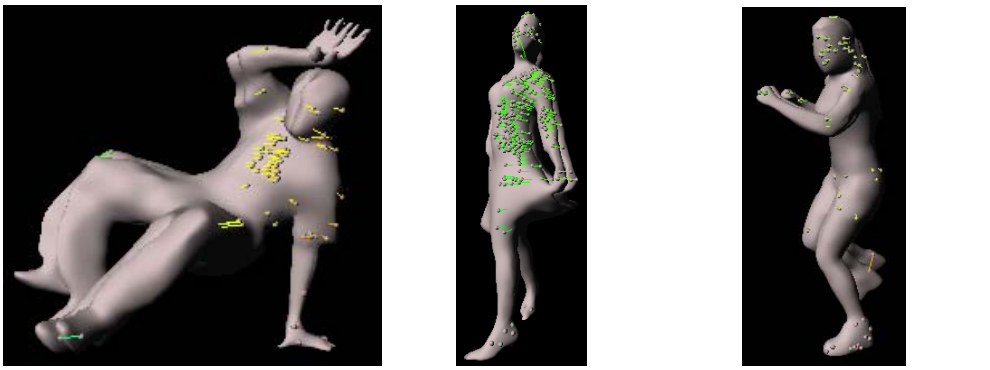


Figure 9: Characteristics of SIFT detection/tracking on multiple view video sequences for different motions.



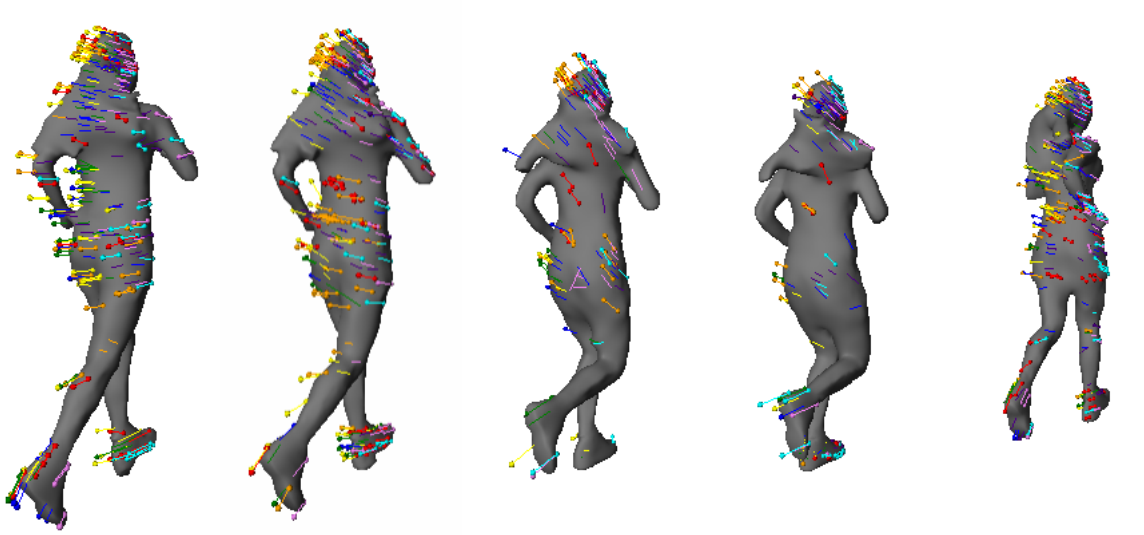
(a) JP Free Dance (b) Roxanne Twirl (c) Roxanne Gamecharacter

Figure 10: 3D single view video sequence matching results using SIFT.

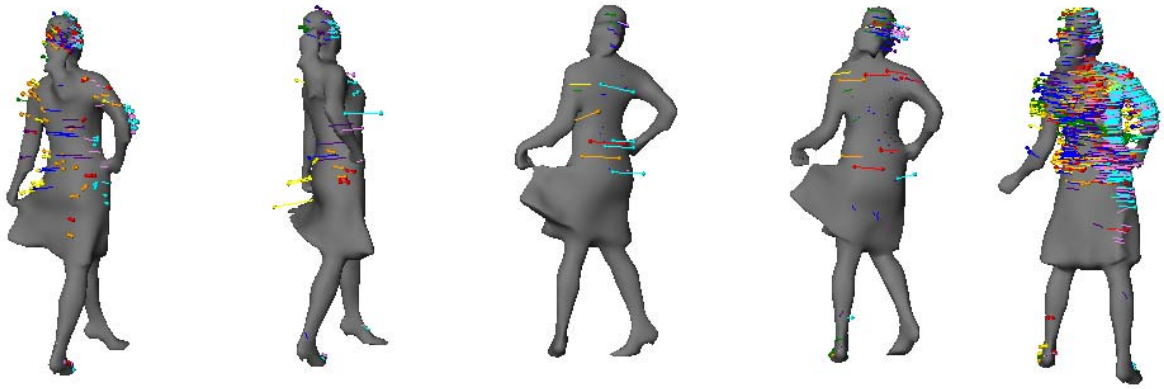
meshes/frames. These links can be used for tracking of features from frame to frame in a video sequence as is shown in Figure 11. Figure 11(a) shows a sample of mesh frames for the Jog sequence and Figure 11(b) is for the Twirl sequence. The links are features matched in 2D texture space in a specific camera view and is colour coded in the following manner: (i) Camera 0 - red, (ii) Camera 1 - orange, (iii) Camera 2 - yellow, (iv) Camera 3 - green, (v) Camera 4 - blue, (vi) Camera 5 - indigo, (vii) Camera 6 - violet and (viii) Camera 7 - cyan. From these figures, our understanding of different motion characteristics and problems faced for feature selection and matching during motion or non-rigid surface deformation is increased. The feature matches can also be used as a starting point to compute sparse-to-dense correspondences of the whole mesh which can be warped for mesh animation purposes.

6 Conclusions

In this paper, we have presented an evaluation of six main feature/region detectors SIFT, SURF, SS, HesAff-GLOH, HesAff-SIFT and MSER-SIFT for use on video sequences with non-rigid surfaces. The aim of this work is to identify the detector that will per-



(a) Jog Sequence



(b) Twirl Sequence

Figure 11: 3D multiple view video sequence tracking results using SIFT.

form best with moving people from video sequences. The SIFT method has been shown to be robust for affine-invariant feature detection on video sequences of moving people and to consistently outperform the other detectors evaluated. Though SIFT is not the most computationally efficient it is significantly better than the others for both single and multiple view detection and matching. Region detector such as MSER is better suited to detecting a small number of regions which are important such as hands and feet where features are difficult to obtain and subsequently matched. It is envisaged that joint sparse feature and region matching can be utilised to obtain good pose estimation which can then be used to obtain dense correspondences for video based animation of people. Also worth noting is that motion phases in a sequence can be characterised to have either no motion, constant motion or fast motion by using feature matching.

7 Acknowledgments

This work is supported by EPSRC grant EP/E001351 'Video-based animation of people'.

References

[AdAT+05] N. Ahmed, E. de Aguiar, C. Theobalt, M. Magnor, and H.-P. Seidel, *Automatic generation of personalized human avatars from multi-view video*, Proceedings of the ACM symposium on virtual reality software and technology (Monterey, USA), December 2005, pp. 257–260, ISBN 1-59593-098-1.

[ASK+04] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, H. Pang, and J. Davis, *The correlated correspondence algorithm for unsupervised registration of nonrigid surfaces*, Proceedings of the Neural Information Processing Systems (NIPS) Conference, 2004, pp. 33–40.

[ATR+08] N. Ahmed, C. Theobalt, C. Roessl, S. Thrun, and H.-P. Seidel, *Dense correspondence finding for parameterization-free animation reconstruction from video*, Proceedings of Computer Vision and Pattern Recognition, 2008, pp. 1–8, ISBN 978-1-4244-2242-5.

[BBM09] T. Brox, C. Bregler, and J. Malik, *Large displacement optical flow*, Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, June 2009, ISSN 1063-6919, pp. 41–48.

[BTG06] H. Bay, T. Tuytelaars, and L. J. V. Gool, *SURF: Speeded up robust features*, International Conference on Computer Vision and Pattern Recognition, 2006, pp. 404–417, ISBN 978-3-540-33832-1.

[dAST+08] E. de Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun, *Performance capture from sparse multi-view video*, Proceedings of SIGGRAPH, 2008, Article no. 98, ISSN 0730-0301.

[dATSS07] E. de Aguiar, C. Theobalt, C. Stoll, and H.-P. Seidel, *Marker-less deformable mesh tracking for human shape and motion capture*, International Conference on Computer Vision and Pattern Recognition (Minneapolis, USA), June 2007, pp. 1–8, ISBN 1-4244-1180-7.

[DSH08] A. Doshi, J. Starck, and A. Hilton, *An empirical study of non-rigid surface feature matching*, Proceedings of the 5th European Conference on Visual Media Production, November 2008, ISSN 0537-9989, pp. 1–10.

[Faw06] T. Fawcett, *An introduction to ROC analysis*, Pattern Recognition Letters **27** (2006), no. 8, 861–874, ISSN 0167-8655.

[HL03] J. S. Hare and P. H. Lewis, *Scale saliency: applications in visual matching, tracking and view-based object recognition*, Distributed Multimedia Systems 2003 / Visual Information Systems 2003, 2003, pp. 436–440, ISBN 1-891706-13-6.

[HSKK01] M. Hilaga, Y. Shinagawa, T. Kohmura, and T. L. Kunii, *Topology matching for fully automatic similarity estimation of 3D shapes*, SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques (New York, NY, USA), ACM,

2001, pp. 203–212, ISBN 1-58113-374-X.

[KB01] T. Kadir and M. Brady, *Scale, saliency and image description*, International Journal of Computer Vision **45** (2001), no. 2, 83–105, ISSN 0920-5691.

[KS04] Y. Ke and R. Sukthankar, *PCA-SIFT: a more distinctive representation for local image descriptors*, Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on, vol. 2, June-July 2004, ISSN 1063-6919, pp. II–506–II–513 Vol.2.

[KZ07] J. Klippenstein and H. Zhang, *Quantitative evaluation of feature extractors for visual slam*, Proceedings of the Fourth Canadian Conference on Computer and Robot Vision, 2007, pp. 157–164, ISBN 0-7695-2786-8.

[LA08] J. Li and N. M. Allinson, *A comprehensive review of current local features for computer vision*, Neurocomputing **71** (2008), no. 10-12, 1771–1787, ISSN 0925-2312.

[Low04] D. Lowe, *Distinctive image features from scale-invariant keypoints*, International Journal of Computer Vision **60** (2004), no. 2, 91–110, ISSN 0920-5691.

[MCMP02] J. Matas, O. Chum, U. Martin, and T Pajdla, *Robust wide baseline stereo from maximally stable extremal regions*, British Machine Vision Conference, vol. 1, 2002, pp. 384–393.

[MS05] K. Mikolajczyk and C. Schmid, *A performance evaluation of local descriptors*, IEEE Transactions on Pattern Analysis & Machine Intelligence **27** (2005), no. 10, 1615–1630, ISSN 0162-8828.

[MTS+05] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, *A comparison of affine region detectors*, International Journal of Computer Vision **65** (2005), no. 1/2, 43–72, ISSN 0920-5691.

[OFL07] M. Ozuysal, P. Fua, and V. Lepetit, *Fast Keypoint Recognition in Ten Lines of Code*, Conference on Computer Vision and Pattern Recognition (Minneapolis, MI), June 2007, pp. 1–8, ISBN 1-4244-1180-7.

[RKD08] B. Ruf, E. Kokiopoulou, and M. Detyniecki, *Mobile museum guide based on fast sift recognition*, Tech. report, Infoscience, Ecole Polytechnique Federale de Lausanne, Switzerland, 2008.

[SH07a] J. Starck and A. Hilton, *Surface capture for performance based animation*, Computer Graphics and Applications **27** (2007), no. 3, 21–31, ISSN 0272-1716.

[SH07b] J. Starck and A. Hilton, *Correspondence labelling for wide timeframe free form surface matching*, International Conference on Computer Vision, October 2007, pp. 1–8, ISBN 978-1-4244-1631-8.

[SMH05] J. Starck, G. Miller, and A. Hilton, *Video-based character animation*, Eurographics/ACM SIGGRAPH Symposium on Computer Animation, 2005, pp. 49–58, ISBN 1-7695-2270-X.

[SMH06] J. Starck, G. Miller, and A. Hilton, *Volumetric stereo with silhouette and feature constraints*, British Machine Vision Conference, vol. 3, 2006, pp. 1189–1198.

[TLF08] E. Tola, V. Lepetit, and P. Fua, *A Fast Local Descriptor for Dense Matching*, Conference on Computer Vision and Pattern Recognition (Alaska, USA), 2008, ISBN 978-1-4244-2242-5.

[TM08] T. Tuytelaars and K. Mikolajczyk, *Local invariant feature detectors: a survey*, Foundations and Trends in Computer Graphics and Vision (2008), ISSN 1572-2740.

[VBK02] S. Vedula, S. Baker, and T. Kanade, *Spatio-temporal view interpolation*, Rendering Techniques, vol. 28, 2002, pp. 65–76, ISBN 1-58113-534-3.

[VZBH08] K. Varanasi, A. Zaharescu, E. Boyer, and R. P. Horaud, *Temporal surface tracking using mesh evolution*, Proceedings of the Tenth European Conference on Computer Vision (Marseille, France), LNCS, vol. Part II, Springer-Verlag, October 2008, pp. 30–43, ISBN 978-3-540-88685-3.

[ZBVH09] A. Zaharescu, E. Boyer, K. Varanasi, and R. Horaud, *Surface feature detection and description with applications to mesh matching*, Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, June 2009, ISSN 1063-6919, pp. 373–380.

Citation

Ashish Doshi, Jonathan Starck, and Adrian Hilton, *An empirical study of non-rigid surface feature matching of human from 3D video*, Journal of Virtual Reality and Broadcasting, 7(2010), no. 3, August 2010, urn:nbn:de:0009-6-25740, ISSN 1860-2037.