

# Head pose estimation for wearable robot control

B. Tordoff, W.W. Mayol, T.E. de Campos and D.W. Murray  
Department of Engineering Science, University of Oxford  
Parks Road, Oxford OX1 3PJ, UK

## Abstract

Recent advances in wearable sensing allow active control of the orientation of a body-mounted camera worn by a remote user. In this paper we consider the control of the active camera from head movements. In the context of tele-operation, these may be the head movements of a remote operator, perhaps acting as the wearer's assistant. The movements are likely to be larger than those in video-conference applications, and so frontal facial features are insufficient. The paper presents a model which incrementally combines a fixed 3D shape model with specific features found on the observed head. Robust methods, including the incorporation of a colour model, are used to mitigate the effect of mismatching, the main contribution being the use of both interest point and colour features within a single random-sampling framework.

## 1 Introduction

The application of robotics in the personal and wearable domain requires sensors and actuators to be responsive to the posture and actions of both the wearer and those in the wearer's surroundings. In recent work [9, 10] we have developed a miniature wearable active camera, attached to a collar and worn at shoulder height. Our device can operate independently of the wearer, either teleoperated or autonomously, and differs therefore from the static shoulder-mounted camera of Pentland et al. [12] which was designed to recover visual "ambience", and differs from cameras attached to the wearer's head, chest or arms [16, 15, 11, 13] which are locked to the wearer's attention.

In [9] the camera's gaze direction was controlled closed-loop from inertial and inside-out visual sensing to stabilise against gross movements of the wearer, and open-loop to switch between visual contexts. In this paper we consider camera control from observing head movements. These may be the head movements of a remote operator, giving a controllable view of the wearer's environment (Figure 1), or of the wearer himself, aligning the camera with the wearer's direction of attention.

Markerless tracking of the face or head from a single viewpoint is commonly achieved using pre-defined features, often including the pupils (eg [4]) or corners of the eyes (eg

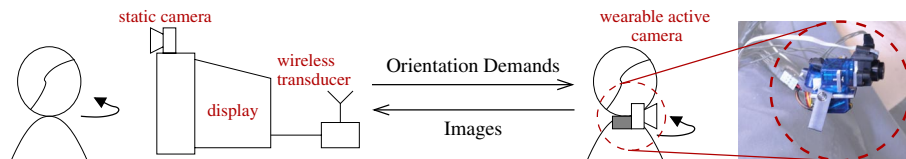


Figure 1: Outside-in vision is used to estimate supervisor head orientation (left) which is sent as a demand to the wearable robot (right). The wearable robot images are then returned to the supervisor.

[5, 19]), corners of the mouth (eg [4, 5, 19]) and nostrils (eg [3]). The results of these papers indicate that such distinctive features are both repeatable and localisable for most individuals over a finite range of rotations, but that they can easily be obscured in those wearing glasses, moustaches, beards, etc., and are quickly lost as the view moves away from frontal. Few, if any, approaches using a small number of facial features can handle head rotations approaching  $90^\circ$ .

A more robust approach to recovering large head rotations is to project the head onto an ellipsoid or cylinder, and to train the model from views representing large panning rotations and substantial elevations, eg. [8, 20]. The approach in [20] is interesting in that instead of intensity values, it maps the response of an interest operator at positions around the whole head. However, despite using an interest operator, the average orientation error in the method is some  $10^\circ$ .

Here we increase the accuracy of such methods using better feature placement and robust guided matching. We address the problems of background clutter and re-initialisation using a method based on colour, and demonstrate how to combine the two.

## 2 Head tracking from interest points

Wu and Toyama [20] model the human head as a set of regularly spaced 3D points on an ellipsoid. During training, the model is manually aligned with a set of views, the 3D points projected into the image, and an interest-point operator run at the projected position to generate a signature response. Any interest-point detector could be used, and Wu and Toyama test a combination of Gabor filters and Laplacians. In this work we will use the Harris-Stephens operator [6] which has the advantages of isotropism, an ability to provide repeatable features over a range of distortions [14], and modest computational cost.

Our first aim is to increase accuracy, and so instead of choosing the model points uniformly *before* viewing the images, the images are used to help generate the model. Response peaks are located to sub-pixel accuracy before projecting onto the model. With little additional calculation the operator is run at a range of scales forming octaves in scale-space, and all peak responses stored. This gives a sparse map of features, but concentrated in areas where the head structure or texture will allow good localisation.

### 2.1 Building the model

Building our model involves the following steps.

**Step 1.** In the first training image, the underlying 3D head model is aligned manually, using pre-defined model points corresponding to corners of the eyes, mouth and the ear-holes. The underlying 3D shape is not an ellipsoid, but a triangular facet head model provided by the Human Animation Working Group [1]. The model should ideally be deformed to better match the subject, but we find that the marginal improvement in tracking performance does not justify the cost in doing so.

Features are detected in the images at scale octaves and those outside the projection of the model boundary eliminated. Those remaining are back-projected onto the appropriate triangular facet assuming affine projection (so that ratios of distances are preserved) and their location stored along with a small bi-linearly interpolated patch of the surrounding image.



Figure 2: Multi-scale interest points detected in the image region underlying the projected model.

Figure 2 shows the set of recovered features for views of one subject, the size of square indicating the scale. The corners of mouth and eye are selected as expected, but so too are more user-specific features on the nose, chin and hairline. Some of these features will be spatially unstable due to repeated texture or non rigid motion (eg. hair), and this becomes apparent as more training images are used.

**Step 2.** After detection of image features in each further training image, 3D point features already stored on the model are projected into the image and matches sought. Match scores are evaluated using zero-normalised cross-correlation, and where both the positional error and match-score are within pre-defined thresholds the match is accepted. Any unmatched image points within the head boundary are treated as new features and are back-projected onto the model and stored as in step 1.

**Step 3.** When all the training images have been processed, the effectiveness of each model feature is assessed as the proportion of the training views in which it was successfully detected and matched out of all views in which it would be expected to be visible. The visibility is determined by considering the viewing direction when the feature was stored and comparing to the current viewing direction. Features viewed at less than  $15^\circ$  from their original direction are considered visible. If a feature is no longer counted as visible a new feature may be detected at the same model position, storing a new image-patch so that distorting the patches for matching is unnecessary.

## 2.2 Obtaining pose from the built head model

Recovering the head pose from a new image involves matching the learnt model points with newly detected image features. Potential matches are searched for in a restricted window around the projected position of each visible model point. However, the large set of interest points (typically  $\sim 200$ ) scattered about the head model increases the likelihood that mismatching will occur, and so robust matching methods are required using minimal sets of points.

Although rather below the 10:1 range to relief ratio usually cited for affine viewing conditions, we find a 4-point weak-perspective solution for pose preferable to a 3-point perspective solution. Assuming zero skew, registered image points  $\tilde{\mathbf{x}}_i$  are related to registered model features  $\tilde{\mathbf{X}}_i$  by

$$\tilde{\mathbf{x}}_i = \frac{1}{Z_{av}} \begin{bmatrix} f_x & 0 \\ 0 & f_y \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \end{bmatrix} \tilde{\mathbf{X}}_i = \mathbf{M}\tilde{\mathbf{X}}_i .$$



where  $Z_{av}$  is the average depth of the model points in the camera frame,  $f_x$ ,  $f_y$  the magnifications in  $x$  and  $y$  directions, and  $R_{ij}$  elements of the rotation between model and camera frames.

To calculate the projection  $M$ , at least four model-image correspondences are required. Therefore, at each iteration,  $h$ , of the random sampler the minimal set of four correspondences are chosen and stacked into matrices

$$\mathbf{x}_h = [\tilde{\mathbf{x}}_1 \quad \tilde{\mathbf{x}}_2 \quad \tilde{\mathbf{x}}_3 \quad \tilde{\mathbf{x}}_4] \quad \mathbf{X}_h = [\tilde{\mathbf{X}}_1 \quad \tilde{\mathbf{X}}_2 \quad \tilde{\mathbf{X}}_3 \quad \tilde{\mathbf{X}}_4]$$

and the motion  $M_h$  recovered using the pseudo-inverse  $M_h = \mathbf{x}_h \mathbf{X}_h^\top (\mathbf{X}_h \mathbf{X}_h^\top)^{-1}$ . The remaining correspondences,  $j$ , are normalised by the same mean positions as the sample points and the projected model point  $M_h \tilde{\mathbf{X}}_j$  compared to the image point  $\tilde{\mathbf{x}}_j$  to evaluate the veracity of the proposed projection.

Rather than choosing the four samples at random, recent extensions [17] to Torr and Zisserman's MLESAC algorithm [18] are used to provide guided sampling and consensus. Under the hypothesis that the motion estimate  $M_h$  is correct, we assume that any positional error  $r_j$  between the projected model point and the image point must be due to Gaussian noise with deviation  $\sigma$  if the match  $j$  is valid ( $v_j$ ). If the match is not valid ( $\bar{v}_j$ ), then the probability of observing a particular positional error is assumed uniform, giving

$$p(r_j | M_h, v_j) = \frac{1}{2\pi\sigma^2} e^{-\frac{r_j^2}{2\sigma^2}} \quad \text{and} \quad p(r_j | M_h, \bar{v}_j) = \frac{1}{A} .$$

where  $A$  is the area searched for the match. The likelihood of a particular positional error given the motion is therefore a mixture

$$p(r_j | M_h) = \left( \frac{1}{2\pi\sigma^2} e^{-\frac{r_j^2}{2\sigma^2}} \right) p(v_j) + \left( \frac{1}{A} \right) (1 - p(v_j)) .$$

Making the further assumption that the positional error for each feature is independent of the error in the other features then the log-likelihood of the entire set of errors  $R_h$  is

$$\log p(R_h | M_h) = \sum_j p(r_j | M_h)$$

The selection of samples, motion calculation and likelihood evaluation are repeated a large number of times, and the motion hypothesis with highest likelihood selected. A final motion estimate is then performed using all features which are likely to be correctly matched, ie. for all  $j$  where  $p(r_j | M_h, v_j)p(v_j) > p(r_j | M_h, \bar{v}_j)p(\bar{v}_j)$ . The priors  $p(v_j)$  and  $p(\bar{v}_j)$  are determined from the match correlation score, and are also used to weight the random sampling such that features which are more likely to be correctly matched are selected preferentially [17]. This greatly reduces the typical number of iterations required. Some typical samples, their support and pose estimate are shown in figure 3.

### 2.3 Tracking performance using interest points

Figure 4 shows results for tracking using only interest points. The model is aligned manually in the first frame, after which tracking progresses successfully up until frame 120. After this point some model features match the background and tracking fails gracefully



Figure 3: The sample matches (red squares) used to calculate a motion hypothesis and the other matches (black crosses) which support the hypothesis. The likelihood of the pose increases from left to right, the model showing the final pose hypothesis.

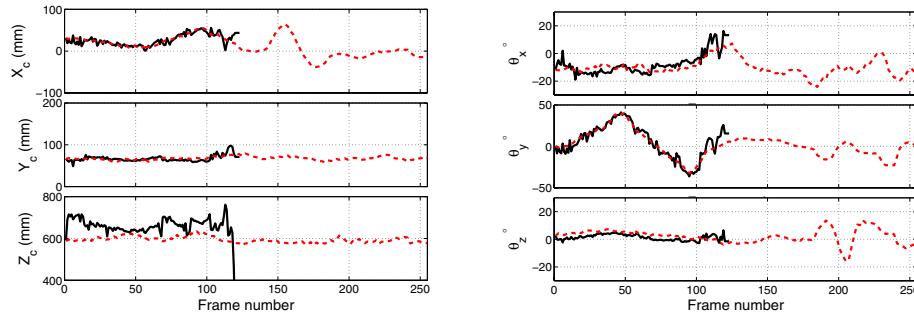


Figure 4: Tracking using features follows both translational and rotational motions well until frame 120, where the tracker fails gracefully. The dashed line shows the manually aligned ground truth, and the solid black line the tracker estimates.

(this is most obvious in the  $Z$  component of the model position). Even using a restricted search window for matching and robust motion estimation, it is found that the tracker is not robust *enough* — once it begins to lose track of the head, failure is swift and permanent.

Re-initialisation using interest points requires a computationally expensive search through either the pose space or the space of possible matches. Instead we explore whether a colour-based approach could better constrain head position and provide a method for fast (re-)initialisation.

### 3 Head tracking from colour

Finding faces in images is routinely achieved by searching for skin-coloured blobs, eg. [19, 3, 2], but use of a single colour distribution for the entire head cannot, of course, yield orientation. Here, we store histograms for individual positions on the head, encoding the colour variations as the orientation changes.

#### 3.1 Acquiring the head colour model

A set of 3D locations from the entire surface of the faceted head model (the facet vertices) is selected for storing colour features in  $YC_rC_b$  space. The model is manually aligned with each training image (as for the feature-based method), and potentially visible model points projected into the image. Visibility is determined using the angle between the

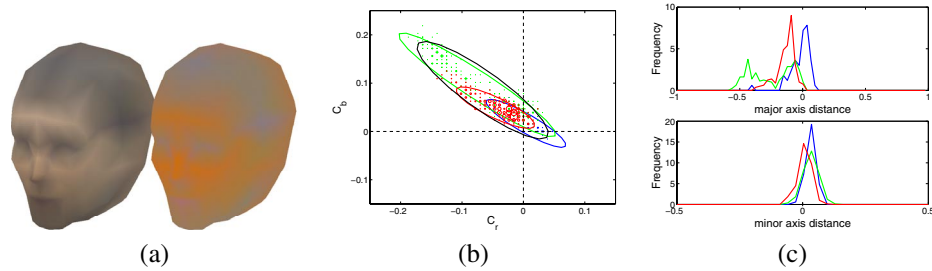


Figure 5: (a) A trained colour model shown in full colour (left) and just the chroma components (right). (b) Three features from a model trained on over 100 images are shown in different colours together with their covariance ellipses, the overall distribution shown in black. (c) The distributions mapped onto the major and minor axes of the overall distribution

model surface normal and the viewing direction. The image colours underlying the projected model points are found by bi-linear interpolation of the neighbouring pixels and stored on the model. In each training view the process is repeated, and the mean and covariance of the colour for each model point derived. Combining the measurements over all features gives an overall distribution for the whole head similar to that used by Birchfield [2].

Figure 5(a) is an example of an acquired model and Figure 5(b) shows a 2D histogram and covariance ellipses of the chromacity for three typical features each sampled more than 100 times together with the overall head distribution. Note that the major and minor axes of the distributions for the individual features are broadly aligned, allowing each feature to be represented by two 1D Gaussians aligned with the overall distribution's axes as in figure 5(c). Furthermore, the distributions along the minor axis do not differ significantly from the overall distribution<sup>1</sup>.

If a significant number of training observations are made, the distribution specific to the feature should be used. However, if the number of observations of a point is zero or small, it is possible to mix the distributions of feature-specific and overall head distributions. For a feature with  $N$  observations

- if  $N = 0$ , use the overall head distribution.
- if  $N = 1$ , calculate the first moment of the feature's colour distribution  $\mu_{\text{feature}}$ . Mix this with the first moment of the head distribution  $\mu_{\text{head}}$  using the rule  $\mu_{\text{new}} = \alpha\mu_{\text{feature}} + (1 - \alpha)\mu_{\text{head}}$ . Second and higher moments (if used) are taken from the head distribution alone.
- If  $N > 1$ , also calculate the second moment for the feature. Using the same mixing weight  $\alpha$ , the covariance is the weighted "mean of variances" plus "variance of means",  $C_{\text{new}} = \alpha C_{\text{feature}} + (1 - \alpha)C_{\text{head}} + \alpha(1 - \alpha)(\mu_{\text{feature}} - \mu_{\text{head}})^2$ .

The weight term  $\alpha$  should have the property that for  $N = 0$ ,  $\alpha = 0$ , and for  $N > 5$ , say,  $\alpha \rightarrow 1$ . Here we use  $\alpha = 1 - e^{-N/2}$ . Figure 6 shows how the feature-specific and head distributions are mixed for one feature given an increasing number of observations.

<sup>1</sup>if one considers that the angle in chromacity space is related to *hue* and the radial distance to *saturation* this result merely shows that the variation between features in saturation is larger than in hue, and is not unexpected.

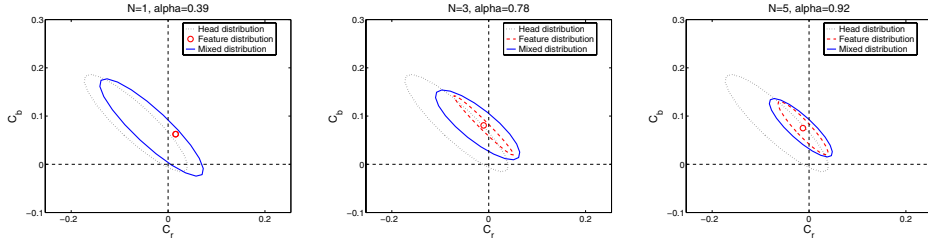


Figure 6: The covariance of the feature observations (red dashed) and overall head (black dotted) are mixed according to the number of observations  $N = 1, 3, 5$  (blue solid).

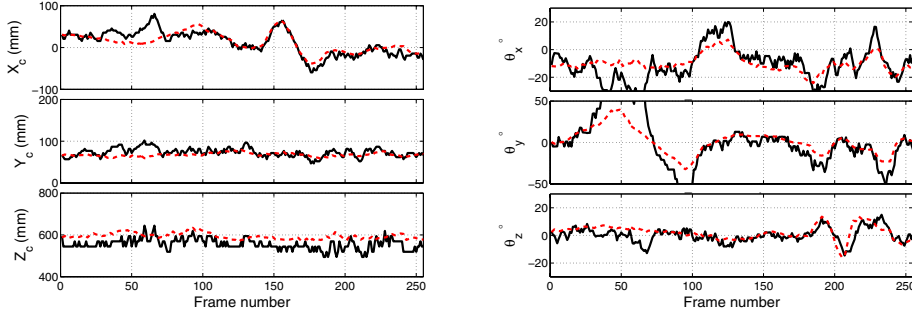


Figure 7: Tracking using only colour information. A local search of the parameter space around the last known pose is sufficient to recover the head translation well, but the rotation is inaccurate and noisy.

### 3.2 Tracking by colour alone

In order to find or track the head using the stored colour information, we hypothesise a pose  $P$ , project the visible model points into the image and determine  $p(c_i|P)$ , the likelihood of each underlying image colour  $c_i$ . A simplifying assumption is that the colour distributions are Gaussian, and that observations for different points are independent so that the overall probability of the observations  $C$  given the pose  $P$

$$p(C|P) = \prod_i p(c_i|P) = \prod_i \frac{1}{2\sigma_{hi}\sigma_{si}\pi} \exp -\frac{1}{2} \left[ \left( \frac{c_{si} - \mu_{si}}{\sigma_{si}} \right)^2 + \left( \frac{c_{hi} - \mu_{hi}}{\sigma_{hi}} \right)^2 \right]$$

where  $c_{si}$  and  $c_{hi}$  are the image chromacity expressed along the overall head distribution's major and minor axes respectively.

Figure 7 shows the pose recovered by colour information alone for the same sequence as figure 4. The model is aligned manually in the first frame, and in successive frames the most likely pose is selected by exhaustively evaluating poses in a small volume around the current pose parameters. With six parameters, just five increments in each parameter results in measuring the likelihood of over 15,000 hypotheses (taking around 4 seconds on a 1.8GHz Pentium IV) — clearly unacceptable for real-time tracking. Even with this scale of search the orientation of the head is badly estimated, although the head is still localised well over the whole sequence (seen from the translation components).

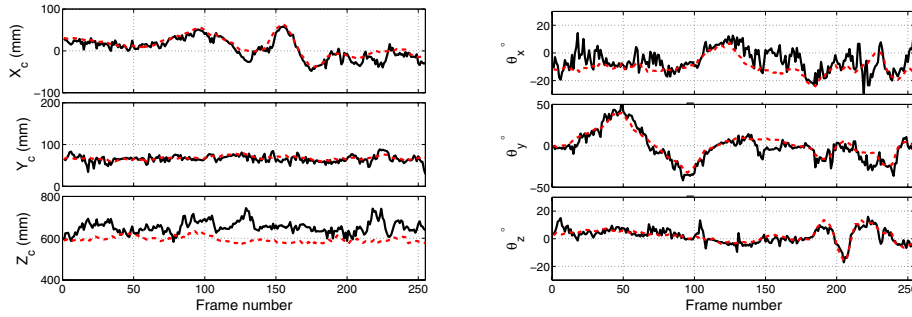


Figure 8: Tracking using the combined method. Interest points are used to hypothesise motions which are then tested using both interest points and the colour features.

## 4 Tracking from interest points and colour

In the previous two sections we have seen that with interest points background clutter is a major problem, but motion hypotheses can be created and their support measured in a fast and meaningful way; whereas using colour the orientation is only weakly constrained but the translational information is sufficient to ensure that the head boundary is well localised. A suitable conflation of the two methods produces a tracker with accuracy and robustness.

Both methods have been described in terms of likelihoods and it is a simple matter to combine them. Interest points are used to hypothesise new poses in the guided MLE-SAC algorithm, and the likelihood of the pose evaluated using both the remaining interest points and all visible colour features. Assuming that interest point positional errors  $r_i$  forming the set  $R$  and colour measurements  $c_j$  from the set  $C$  are independent, the likelihood of the hypothesised motion  $M_h$  is

$$p(C, R|M_h) = p(C|M_h)p(R|M_h) = \prod_i p(r_i|M_h) \prod_j p(c_j|M_h)$$

In cases where not enough interest points are available, or where the motions hypothesised are all outside sensible limits, a small search is made using colour alone. This can prevent outright loss of tracking in some cases. In the first frame and when tracking is lost a coarse global search using colour reinitialises the pose automatically.

Figure 8 shows the results for the combined tracker on the same sequence as figures 4 and 7. The interest points provide a convenient way to hypothesise motions, and the colour information helps to reject motions that would place the head over the background. At each new frame guided MLESAC was evaluated for 100 samples, the complete algorithm taking less than 50ms on a 1.8GHz Pentium IV. This is sufficient for 20Hz tracking, but with optimisation it appears feasible to reduced this below 40ms for video-rate tracking.

The accuracy of the three methods is compared in the table below. Bias is indicated by a non-zero mean value, and uncertainty by the  $\pm$  value. These accuracies compare favourably with those reported by Wu and Toyama ( $> 10^\circ$ ), although direct comparison on extended sequences is required to give a fairer comparison. The combined method is nearly as accurate as the interest point method, but gains robustness from colour.

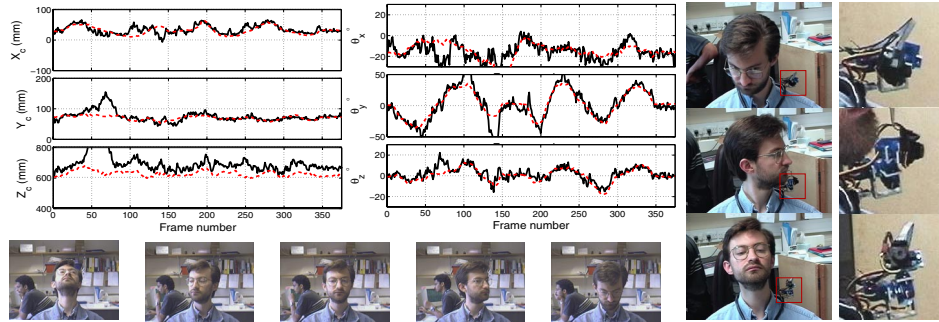


Figure 9: Tracking results for a second victim, using a model trained from seven images, five of which are shown along the bottom. Three snapshots of the supervisor head orientation and slaved robot orientation are shown to the right.

	Interest points only	Colour only	Combined
$X_c$ (mm)	1.01 $\pm$ 9.06	-0.39 $\pm$ 15.70	-6.79 $\pm$ 8.73
$Y_c$ (mm)	0.21 $\pm$ 6.34	5.41 $\pm$ 10.34	-1.56 $\pm$ 7.53
$Z_c$ (mm)	41.39 $\pm$ 98.60	-37.89 $\pm$ 23.87	60.21 $\pm$ 30.74
$\theta_x$ ( $^\circ$ )	0.92 $\pm$ 4.55	-0.62 $\pm$ 8.78	4.16 $\pm$ 6.93
$\theta_y$ ( $^\circ$ )	1.22 $\pm$ 6.11	-1.58 $\pm$ 15.03	-3.00 $\pm$ 6.62
$\theta_z$ ( $^\circ$ )	-1.81 $\pm$ 2.79	-2.05 $\pm$ 4.45	-0.10 $\pm$ 2.76

The final results in figure 9 show the pose estimates for a second sequence trained from seven views of a different user, five of which are shown. The recovered orientation is low-pass filtered before being sent to the wearable device. Of particular importance is that during periods where pose is not accurate (eg. frames 40–70) the colour information prevents the model drifting onto the background, ie. the  $X_c$ ,  $Y_c$  errors remain small.

## 5 Discussion and conclusions

In this paper we have demonstrated that back-projecting interest points specific to the viewed subject onto an underlying pre-imposed 3D head shape provides a method for building a head model. Such a model can determine head pose over a wide angular range and improves over time, provided it never includes points from the background.

Although robust methods using guided MLESAC reduce mismatching, inadvertent matching of background points has dire consequences for pose recovery. We thus proposed running a colour model alongside, with the aim of providing good localisation and some orientation information. Much remains to be done to characterise experimental performance, but the combination has been demonstrated to function on “difficult” spectacled and bearded subjects and with significantly non-uniform lighting. The attempt to gain orientation information from the colour model by considering the distributions of individual features can be improved by continuing to refine the model during pose tracking. At present the number of explicit training images is small, and so the individual distributions are dominated by the overall distribution.

At present the output is filtered but tracking is not. Our experience is that considerable care is required here. In [7], measurements of pose over many minutes suggest that muscular action in the neck should be modelled as an Ornstein-Uhlenbeck stochastic process



when creating the pose tracking filter. We aim to use such a filter for this application.

In this paper we have focussed on controlling an active head to obtain a real sense of a remote environment, however mixing interest point and colour information as shown is more generally applicable. Other information expressed in a probabilistic manner might also be included, such as background subtraction and occluding contour detection.

## Acknowledgements

This research is supported by UK EPSRC Grant GR/N03266, by a Mexican CONACYT scholarship to WWM and by a Brazilian CAPES scholarship to TEdeC.

## References

- [1] The human animation working group: [www.h-anim.org](http://www.h-anim.org).
- [2] S Birchfield. Elliptical head tracking using intensity gradients and color histograms. In *Proc of the IEEE Conf on Computer Vision and Pattern Recognition*, June 1998.
- [3] R.S. Feris, T.E. de Campos, and R.M. Cesar. Detection and tracking of facial features in video sequences. In *Mexican International Conference on Artificial Intelligence*, volume 1793 of *Lecture notes in Artificial Intelligence*, pages 197–206, 2000.
- [4] A Gee and R Cipolla. Determining the gaze of faces in images. *Image and Vision Computing*, 12(10):639–647, 1994.
- [5] S.B. Gokturk, J-Y. Bouguet, and R. Grzeszczuk. A data-driven model for monocular face tracking. In *Proc 8th Int Conf on Computer Vision, Vancouver*, 2001.
- [6] C. G. Harris and M. Stephens. A combined corner and edge detector. In *Proc 4th Alvey Vision Conf, Manchester*, pages 147–151, 1988.
- [7] J.J. Heuring and D.W. Murray. Modeling and Copying Human Head Movements. *IEEE Transactions on Robotics and Automation*, 15(6):1095–1108, 1999.
- [8] M. la Cascia and S. Sclaroff. Fast, reliable head tracking under varying illumination. In *Proc of the IEEE Conf on Computer Vision and Pattern Recognition*, pages 604–610, 1999.
- [9] W.W. Mayol, B. Tordoff, and D.W. Murray. Wearable visual robots. In *Int Symposium on Wearable Computing*, 2000.
- [10] W.W. Mayol, B. Tordoff, and D.W. Murray. Designing a miniature wearable visual robot. In *Proc IEEE Int Conf on Robotics and Automation*, 2002.
- [11] N. Kohtake, J. Rekimoto, and Y. Anzai. Infostick: an interaction device for inter-appliance computing. In *In Proc. Workshop on Handheld and Ubiquitous Computing*, pages 246–258, September 1999.
- [12] A. Pentland. Looking at people: Sensing for ubiquitous and wearable computing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1), January 2000.
- [13] S. Mann. Wearcam (the wearable camera). *Proc. IEEE Int. Symp. on Wearable Computing*, October 1998.
- [14] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. *Int. Journal of Computer Vision*, 37(2):151–172, 2000.
- [15] T. Starner, J. Auxier, D. Ashbrook, and M. Gandy. The gesture pendant: a self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. In *IEEE International Symposium on Wearable Computing*, August 2000.
- [16] T. Starner, J. Weaver, and A. Pentland. Real-time american sign language recognition using desk and wearable computer based video. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(12):1371–1375, December 1998.
- [17] B. Tordoff and D.W. Murray. Guided sampling and consensus for motion estimation. In *Proc 7th European Conf on Computer Vision, Copenhagen*, June 2002.
- [18] P. H. S. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78:138–156, 2000.
- [19] K. Toyama. Prolegomena for Robust Face Tracking. In *ECCV Workshop on Advances in Facial Image Analysis and Recognition Technology*, May 1998.
- [20] Y Wu and K Toyama. Wide-range, person- and illumination-insensitive head orientation estimation. In *International conference on Face and Gesture Recognition*, pages 183–188, 2000.