Real-time non-rigid surface detection

Julien Pilet, Vincent Lepetit, Pascal Fua

Computer Vision Laboratory

École Polytechnique Fédérale de Lausanne, Switzerland

Email: {Julien.Pilet, Vincent.Lepetit, Pascal.Fua}@epfl.ch

http://cvlab.epfl.ch/

November 3, 2004

Abstract

We present a real-time method for detecting deformable surfaces, with no need whatsoever for a priori pose knowledge.

Our method starts from a set of wide baseline point matches between an undeformed image of the object and the image in which it is to be detected. The matches are used not only to detect but also to compute a precise mapping from one to the other. The algorithm is robust to large deformations, lighting changes, motion blur, and occlusions. It runs at 10 frames per second on a 2.8 GHz PC and we are not aware of any other published technique that produces similar results.

Introducing deformable meshes, along with a well designed robust estimator, is the key to dealing with the large number of parameters involved in modeling deformable surfaces and rejecting erroneous matches for error rates of up to 95%, which is considerably more than what is required in practice.

1 Introduction

Rigid object detection and tracking have been extensively studied and effective, robust, and real-time solution proposed [30, 21, 20, 23]. The two are of course complementary since trackers require initialization and, no matter how good they may be, will sometimes lose track, for example, because of severe occlusions. Non-rigid object tracking has also been convincingly demonstrated, for example in the case of animated faces [8, 7, 2] or even more generic and deformable objects [3]. However, the automated detection of such deformable objects still lags behind and existing methods [5, 10] are far less convincing for real-time applications. They tend to be computationally intensive and are usually geared more towards recognition or segmentation than providing the kind of fast initialization that a tracker needs to recover from potential failures.

In this paper, we propose a method that fits this requirement by allowing fast and robust detection and registration of an object that can be subjected to very large non-affine deformations such as the piece of foam of Fig. 5. It relies on wide-baseline matching of 2–D feature points, which makes it resistant to partial occlusions and cluttered backgrounds: Even if some features are missing, the object can still be detected as long as enough are found and matched. Spurious matches are removed by enforcing smoothness constraints on the deformation, which is done very quickly in our approach.

More specifically, at the heart of our approach is a very fast wide-baseline point matching technique that allows us to establish correspondences between keypoints extracted from a training image of the undeformed object to those that can be found when the object deforms [19, 20]. Given such correspondences, if the target object were rigid, detecting it and estimating its pose could be implemented using a robust estimator such as RANSAC [11]. However, for a deformable object, the problem becomes far more complex because not only pose but also a large number of deformation parameters must be estimated.

The main contribution of this paper is the introduction of deformable 2–D meshes, along with a well designed robust estimator, as the key to deal with this large number of parameters. The keypoints positions are expressed as weighted sums of the mesh vertices in the model image and change as the mesh is deformed. Fitting then amounts to minimizing a criterion that is the sum of two terms. The first is a robust estimate of the square distances of the keypoints in the model image to that of the corresponding ones in the input image. The second is a quadratic deformation energy [12]. As was the case for the original snakes [18], this quadratic term allows the use of a semi-implicit minimization scheme that converges even when the initial estimate is very far from the solution, which, in our context, is what happens when the object is severely deformed. When combined with an appropriately defined robust estimator for

the keypoint distances and optimization schedule, this approach to minimization allows detection in under 100 milliseconds on a 2.8 GHz desktop while being robust to large deformations, severe occlusions, and changes in lighting. In fact, we have verified that our method keeps on working with 95% of point matches being erroneous, which is key to robustness because no real-time matching technique can be expected to work perfectly well in the presence of clutter, orientation changes, shadows, or specularities. We do not know of any other technique able to produce similar results.

In the remainder of the paper, we first review briefly the existing literature and present an overview of our algorithm. We then discuss its most critical steps and present our results.

2 Related work

Many approaches to registering a model on an image have been proposed. Some feature-based algorithms first establish correspondences and then find the best transformation explaining them, while eliminating outliers. Others simultaneously solve for both correspondence and registration, without the need for correspondences and with or without using feature characterization. Finally some techniques do not even rely on features. We review them briefly below and show that these existing approaches have not yet been shown to be suitable for real-time detection of deformable objects.

2.1 Feature-Based Methods

These approaches rely on establishing correspondences between image-features of the target object in one or more images and those that can be found in an input image in which it is to be detected. These correspondences are then used to estimate the transformations.

2.1.1 Establishing Correspondences

For detection purposes, the methods used to extract and match them should be insensitive to viewpoint and illumination changes. Scale-invariant feature extraction can be achieved by using the Harris detector [16] at several Gaussian derivative scales, by considering local optima of pyramidal difference-of-Gaussian filters in scale-space [22], or extremal regions [24]. Mikolajczyck et al. [26] have also defined an affine invariant point detector to handle larger viewpoint changes, but it relies on an iterative estimation that would be too slow for our purposes. Given the extracted feature points, various local descriptors have been proposed: Schmidt and Mohr [28] propose a rotation invariant descriptor that are functions of relatively high order image derivatives to achieve orientation invariance. Baumberg [4] uses a variant of the Fourier-Mellin transformation to achieve rotation invariance. He also gives an algorithm to remove stretch and skew and obtain an affine invariant characterization. Allezard et al. [1] represent the keypoint neighborhood by a hierarchical sampling, and rotation invariance is obtained by starting the circular sampling with respect to the gradient direction. Tuytelaars and al. [29] fit an ellipse to the texture around local intensity extrema and use the Generalized Color Moments [27] to obtain correspondences remarkably robust to viewpoint changes. Lowe [23] introduces a descriptor called SIFT and based on several orientation histograms, that is not fully affine invariant but tolerates significant local deformations.

This last descriptor has been shown in [25] to be one of the most efficient to detect planar objects. However, in our own experience, its performance degrades somehow in the presence of non planar deformations and its computational requirements are too high for our real-time performance requirements.

Shape contexts [5] are an interesting alternative. They are powerful tools for matching shape patterns but are less suited for image registration. Designed to compute a distance between two shapes, they first characterize edge areas on both patterns and then try to establish one to one correspondences using bipartite graph matching. Although this method handles some outliers and slightly different numbers of feature detected on both shapes, it is not ideal to extract objects from a cluttered background.

In practice, we therefore treat wide baseline matching of these keypoints as a classification problem, in which each class corresponds to the set of all possible views of such a point. This formulation [19, 20] gives us access to powerful classification methods to achieve both very fast matching and robustness to non affine deformations.

As will be shown in Section 4, we have actually successfully tested all three approaches to matching – SIFT, classification and shape context characterization – in conjunction with our approach to detection, thus showing that its effectiveness is independent from the specific technique used to establish the point correspondences. However, only the classification tree based technique has proved fast enough for our purpose: real-time without loss of accuracy.

2.1.2 From Correspondences to Detection

Whatever the matching technique used, the correspondences can then be used to detect the object in several different ways.

The simplest is to eliminate outliers and find a globally consistent interpretation using a robust estimator. Having each local match vote for a global transformation is the approach used by the Hough transform and its many variations. This is effective for rigid transformations but impractical for those of deformable objects because they have far

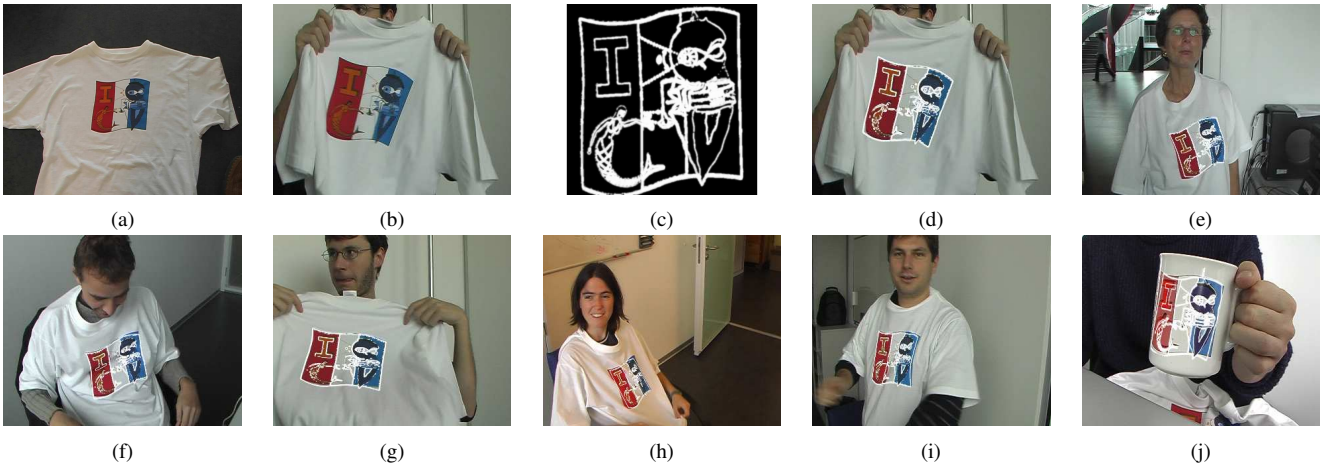


Figure 1: In order to achieve surface detection, we use a model image (a). Then, our method computes a function mapping the model to an input image (b). To illustrate this mapping, we find the contours of the model using a simple gradient operator and we use them as a validation texture (c) which is overlaid on the input image using the recovered transformation (d). Additional results are obtained in different conditions (e to i). Note that in all cases, including the one where the T-shirt is replaced by a cup (j), the white outlines project almost exactly at the right place, thus indicating a correct registration and shape estimation. The registration process, including image acquisition, takes about 100 ms and does not require any initialization or *a priori* pose information.

too many degrees of freedom to discretize the different possible transformations into a vote accumulator. The same can be said of the popular random sample consensus algorithm (RANSAC) [11]: With 25% of outliers and 100 degrees of freedom, 10^{12} samples are required to guarantee with 90% probability of that at least one sample does not contain outliers [17].

An alternative strategy is to proceed iteratively. TPS-RPM (thin plate spline - robust point matching, [6]) and EM-ICP (expectation maximization - iterative closest point, [14, 9]) are two well-known representatives of the family of algorithms that simultaneously solve for both correspondence and transformation using an iterative process. At each step, the current transformation estimate is first used to establish correspondences and assign weights to them, and, then, is refined using those correspondences. These methods use an entropy term—be it called temperature parameter, scale or blurring factor, or variance—that is progressively reduced. It controls the assignment of weights to the correspondences and has an important role in insuring convergence towards a desirable solution. As will be discussed in more detail in Section 3.2, our algorithm follows a similar strategy but makes use of local characterization to reduce the correspondence problem difficulty and to achieve real-time performance.

Image exploration [10] constitutes a third strategy that hooks on a first set of correspondences and then gradually explores the surrounding area, trying to establish more matches. It can handle deformable objects but this complex process is slow and takes several minutes on a 1.4 GHz computer.

2.2 Direct Methods

For objects such as faces whose deformations are well understood and can be modeled in terms of a relatively small number of deformation parameters, fitting directly to the image data without using features is an attractive alternative to using correspondences because it allows the use of global constraints to guide the search. This has been successfully demonstrated in the context of face tracking [8, 7, 2] but this typically requires a good initialization because the criteria being minimized tend to have many local minima. An interesting variation has been proposed recently in [3] where flow is used in conjunction with radial basis functions (RBF) to track objects that are less constrained than faces. However, this approach assumes there is an affine transformation that approximates the deformation well enough to lead RBF centers not too far to their destination so that minimizing an image-based criterion yields the correct answer. This is a strong assumption that may not be correct for a deformation as severe as the one shown in Fig. 5.

By contrast, methods able to automatically detect, as opposed to track, deformable objects are few. A method to instantiate the shape of *applicable* surfaces, such as paper, has been proposed [15] but requires that the whole outline be detected, which severely limits its applicability.

3 Non-rigid Surface Detection

To detect a potentially deformable object, we rely on establishing correspondences between a model image in which the deformations are small and an input image in which they may be large. To this end, we use the fast wide-baseline matching algorithm [19] discussed in Section 2.1.1. Given a set C of correspondences between the two images, some of which might be erroneous, our problem can be formally stated as follows: We are looking for the transformation $T_{\mathbf{S}}$ mapping the undeformed model surface M into the deformed target one $T_{\mathbf{S}}(M)$ and for the subset $G \subset C$ of correct matches such that the sum of the square distances between corresponding points in G is minimized while the deformations remain as smooth as possible.

3.1 2–D Surface Meshes

We represent our model M as a triangulated 2–D mesh of hexagonally connected vertices such as the one shown in Figure 2. The position of a vertex v_j is specified by its image coordinates (x_j, y_j). By computing barycentric coordinates, this representation allows to define a transformation T_S mapping any point on the original model to the transformed mesh, parameterized by the vector $S = (X, Y)$. In other words, we can transform with respect to S a point p on the original surface to

$$T_S(p) = \sum_{i=1}^{3} B_i(p) \begin{bmatrix} x_i \\ y_i \end{bmatrix} ,$$

where $B_i(p)$ are the three barycentric coordinates computed on the original mesh.

The mesh deforms to minimize an objective function $\varepsilon(S)$ whose state vector S is the vector of all x and y coordinates. In practice, we write

$$\varepsilon(S) = \lambda_D \varepsilon_D(S) + \varepsilon_C(S) \ , \tag{1}$$

where ε_C is a data term that takes point correspondences into account, ε_D is a deformation energy that should be rotationally invariant and tend to preserve the regularity of the mesh, and λ_D is a constant. We take $\varepsilon_D(S)$ to be an approximation of the sum over the surface of the square derivatives of the x and y coordinates. Because the mesh is regular, $\varepsilon_D(S)$ can be written using finite differences as

$$\varepsilon_D(S) = 1/2(X^T K X + Y^T K Y) \ , \tag{2}$$

where X and Y are the vectors of the x and y coordinates of the vertices, and K is a sparse and banded matrix [13]. This regularization term serves a dual purpose. First it *convexifies* the energy landscape and improves the convergence properties of the optimization procedure. Second, in the presence of erroneous correspondences, some amount of smoothing is required to prevent the mesh from overfitting the data, and wrinkling the surface excessively. To minimize $\varepsilon(S)$, we use the semi-implicit scheme so successfully introduced in the original snake paper [18]: We are looking for a minimum of the energy and therefore for solutions of

$$\begin{array}{rcl} 0 & = & \frac{\partial \varepsilon}{\partial X} = \frac{\partial \varepsilon_C}{\partial X} + KX \ , \\ 0 & = & \frac{\partial \varepsilon}{\partial Y} = \frac{\partial \varepsilon_C}{\partial Y} + KY \ . \end{array} \tag{3}$$

Since K is positive but not definite, given initial vectors X_0 and Y_0, this can be solved by introducing a viscosity parameter α and iteratively solving at each time step the two coupled equations

$$\begin{aligned} KX_t + \alpha(X_t - X_{t-1}) + \left.\frac{\partial \varepsilon_C}{\partial X}\right|_{X=X_{t-1} Y=Y_{t-1}} &= 0 \ , \\ KY_t + \alpha(Y_t - Y_{t-1}) + \left.\frac{\partial \varepsilon_C}{\partial Y}\right|_{X=X_{t-1}, Y=Y_{t-1}} &= 0 \ , \end{aligned}$$

which implies

$$\begin{aligned} (K + \alpha I)X_t &= \alpha X_{t-1} - \left.\frac{\partial \varepsilon_C}{\partial X}\right|_{X=X_{t-1}, Y=Y_{t-1}} \ , \\ (K + \alpha I)Y_t &= \alpha Y_{t-1} - \left.\frac{\partial \varepsilon_C}{\partial Y}\right|_{X=X_{t-1}, Y=Y_{t-1}} \ . \end{aligned}$$

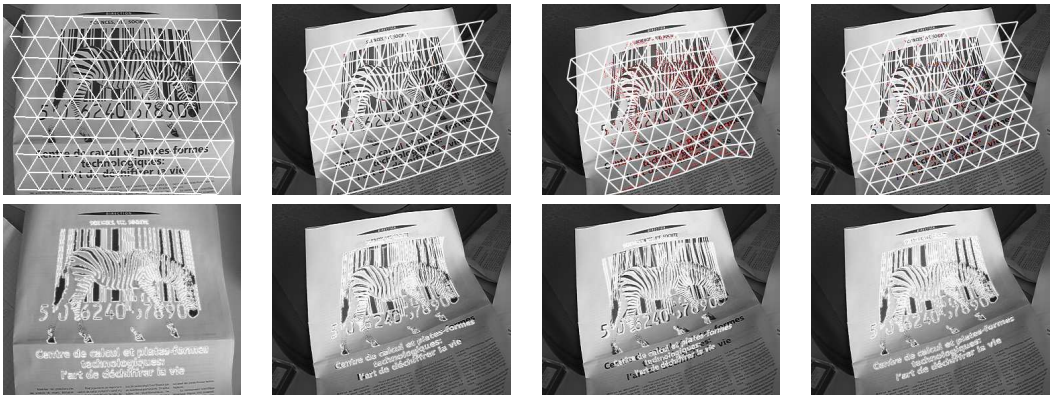
Because K is sparse and regular, solving these linear equations using LU decomposition is fast and upon convergence $X_t \approx X_{t-1}$ and $Y_t \approx Y_{t-1}$. This iterative scheme therefore quickly yields a solution of Eq. 3, even when starting with completely random guesses for X_0 and Y_0 as will be shown in Section 4.

3.2 Correspondence Energy

ε_C, the data term of eq. 1, is designed so that minimizing it tends to reshape the mesh so that it matches the target object in the input image. Let C be a set of *correspondences* between the model and the input image, in which a point can potentially be matched to several ones. For $c = \{c_0, c_1\} \in C$, let c_0 be the 2–D coordinates of a feature point in the model image and c_1 the coordinates of its potential match in the input image. We take the data term to be

$$\varepsilon_C = -\sum_{c \in C} w_c \rho\left(\|c_1 - T_{\mathbf{S}}(c_0)\|, r\right) \ , \tag{4}$$

where ρ is a robust estimator whose *radius of confidence* is r and $w_c \in [0, 1]$ a weight associated to each correspondence. In our experience the choice of ρ is critical to ensure the elimination of outliers and convergence towards the desired minimum while the choice of the w_c has much less impact, as will be discussed below. Note that a particular feature point c_0 in the model image is usually associated to



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

Figure 2: Comparing three different keypoint matching algorithms. (a) Model image and validation texture shown in white. Results using: (b) Real-time classification trees, (c) shape context reimplementation, and (d) SIFT.

several corresponding points in the input image, in which case $c_0$will appear in several elements of C.

Minimizing ε therefore results in a mesh that moves towards the desired solution but whose progression can be blocked by outliers. To overcome this, we introduce a simple optimization schedule in which the initial *radius of confidence* $r_0 = 1000$ is progressively reduced at a constant rate $\eta = 0.5$: $r_t = \eta r_{t-1}$. For each value of r, we minimize ε and use the result as the initial state for the next minimization. When r reaches the noise level expected in the correspondences, around one or two pixels, the algorithm stops. In practice, reducing r 10 times, with 5 mesh optimization iterations each time, proved sufficient for precise registration, which is key to real-time performance.

Once the algorithm has proposed a solution, counting compatible correspondences is a very discriminative measure to know if the solution is correct or not. A simple threshold allows to gracefully handle cases where the surface is completely hidden.

3.2.1 Robust Estimator

We choose $\rho(\delta, r) = \begin{cases} \frac{3\left(r^2-\delta^2\right)}{4r^3} & \delta < r \\ 0 & otherwise \end{cases}$.

As shown in Fig. 3 the shape of ρ is that of a quadratic ridge that gets narrower and taller when r decreases. In other words, r acts as a confidence measure. When it is large, most correspondences, potentially including poor ones, are given some weight. But as r becomes smaller, ρ becomes more peaked and selective.

Note that ρ is normalized so that

$$\int_{-\infty}^{\infty} \rho(x, r)dx = 1 \ \ \forall r > 0 \ \ ,$$

which means that values of ε_C computed using different r values remain commensurate to the $\lambda_D \varepsilon_D$ term of Eq.1. Therefore, we do not need to adjust either the λ_D parameter or the w_c weights of Eq.4. This is in contrast to methods such as SoftAssign[6] in which the surface rigidity must be progressively reduced according to a schedule that is not necessarily easy to synchronize with the annealing of r and may change from case to case.

The quadratic behavior of ρ within the *ridge of confidence* whose size is controlled by r yields a relatively convex ε_C that is easy to minimize. Furthermore, the magnitude ρ's derivatives is inversely proportional to r, which is a desirable behavior: At the beginning when r is large, the gradients of ε_D are comparatively larger than those of ε_C, preventing erroneous matches from crumpling the surface while allowing correct and consistent correspondences to produce the right global deformation. As the process goes on and r decreases, the ρ derivatives and the gradients of ε_C become larger. The triangulation then bends more easily and outliers are rejected.

3.2.2 Disambiguating Multiple Matches

Recall that a point from the model image can have several potential matches in the input image. One can simply rely on the increased tightness ρ function of the previous section to disambiguate those cases as the r parameter decreases or use a more sophisticated weighting scheme. In other words, we can set the the w_c weights of Eq.4 in one of the following ways:

1. $w_c = 1$ for all correspondences,
2. $w_c = 1$ for the closest match, and zero to all others as in ICP,

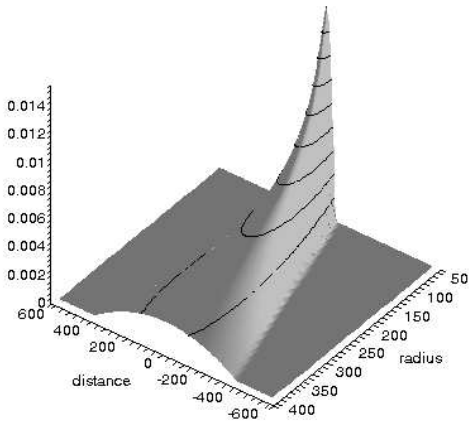


Figure 3: The ρ function of Section 3.2.1 is quadratic for distances smaller than the radius of confidence, elsewhere it is zero.

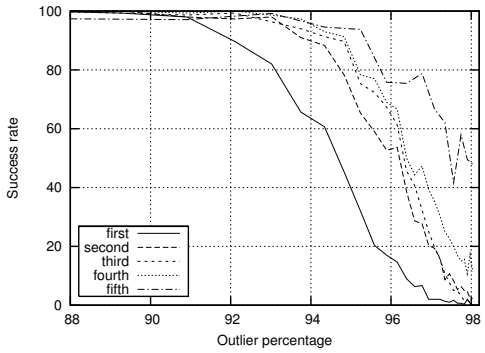


Figure 4: Comparing weighting schemes. Success rate as a function of erroneous correspondences percentage, for each one of the five schemes described in Section 3.2.2.

3. $w_c = \frac{\exp\left(-\|c_1 - T_{\mathbf{S}}(c_0)\|^2/2\sigma^2\right)}{\sum_{d \in C, d_o = c_o} \exp\left(-\|d_1 - T_{\mathbf{S}}(d_0)\|^2/2\sigma^2\right)}$, with $\sigma = \frac{r}{3}$, as in EM-ICP [14],

4. $w_c = \frac{\rho(\|c_1 - T_{\mathbf{S}}(c_0)\|, r)}{\sum_{d \in C, d_0 = c_0} \rho(\|d_1 - T_{\mathbf{S}}(d_0)\|, r)}$, a variation of EM-ICP in which the Gaussian is replaced by ρ,

5. a weight computed by normalizing rows and columns of the correspondence matrix, as in SoftAssign [6].

To compare these weighting schemes, we generated random synthetic data with 500 model and target features, 300 good matches and a variable number of erroneous correspondences. Fig. 4 shows the success rate as a function of the outlier rate. SoftAssign is the only method that takes into account not only the case where a single model point is matched to several destinations, but also the ambiguity of different model points matched to the same destination. Not surprisingly, it yields an improved success rate for very large numbers of outliers but at the cost of a substantial increase in computational complexity. In any event, for outlier rates below 90% which they are in practice, all five weighting schemes are roughly equivalent and we choose the simplest, that is $w_c = 1$, $\forall c \in C$. Note that this is only true because the ρ function does a good job of disambiguating matches. Note also that, here, we have only compared weighting schemes for our specific purposes as opposed to complete implementations.

4 Results

The method has been tested in conjunction with three different feature point recognizers: the publicly available SIFT implementation [23], a reimplementation of shape context characterization [5], and the classification trees-based method [19]. Because our technique is robust, the results are almost indistinguishable, as shown in Fig. 2.

However, because the classification-based method is much faster than the others, it is only when using it that we obtain true real-time performance. In this example, given the optimization schedule of Section 3.2, the algorithm runs at 10 frames per second on a P4 2.8 GHz machine. Because we detect, as opposed to track, we can find objects as soon as they become visible and our method is robust to both perspective distortion and severe deformations. For example, in the example of Fig. 1, the ICCV logo on the shirt is detected very quickly and well before its deformation has become roughly planar. Similarly, the logo is equally well detected when worn by different people or seen on the ICCV mug. Fig. 7 describes similar speed and robustness to deformations when detecting a newspaper page, and Fig. 5 shows detection result on a piece of foam. For well textured objects, we get no false positives and only false negatives when the deformations or occlusions are so severe that the target object is almost impossible to make out, as in the first few frames of Fig. 6. Of course, the performance degrades in the absence of texture and this is one of the issues we will address in future work.

Because the point matcher we use is relatively insensitive to light changes or motion blur, they do not hinder the registration process. Our technique can therefore also be used for Augmented Reality applications, i.e. superimposing an artificial element in a natural video sequence, as shown in Fig. 8. Note that in this example, we used the raw results produced by our real-time algorithm, based only on wide baseline matching. If greater quality were required, it would be easy to refine them by seeking more information from images, using the result of our algorithm as initial transformation.

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

Figure 5: Deforming a piece of foam. (a) Model image and validation texture. (b) to (e) detection results. A video sequence showing this piece of foam is submitted as supplementary material.

Figure 6: A progressively folding and unfolding T-shirt without any false positive detection. As indicated by the symbol on the upper right corner of images, the system knows whether the logo is present or not and overlays the validation texture only in the first case. The full video sequence is submitted as supplementary material.

5 Conclusion

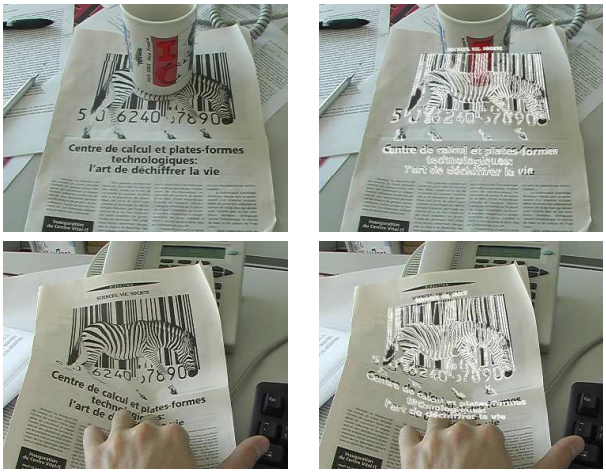
We have demonstrated a very fast and robust approach to detecting deformable surfaces. It is robust to large deformations, changes in lighting, and motion blur and runs at 10 frames per second on a 2.8 GHz PC. It takes advantage of wide-baseline matching, deformable mesh and robust estimation techniques in such a way that the resulting algorithm has very few parameters and they do not require fine tuning.

The current computations are performed using 2–D meshes but the formalism presented in this paper naturally extend to 3–D, with only a very limited additional computational burden. This should be key to handling even more severe self-occlusions than the ones shown in this paper and, also, to incorporate physical knowledge about the deformation modes of the surface if they are known. This should help us handle less textured objects than the ones we have worked with so far, that is objects for which fewer interest points can be detected and matched. An alternative way to deal with relatively bland surfaces would be to broaden the definition of interest points to include those that can be found along contours, as opposed to corners, and could also be considered within our framework. We intend to pursue both avenues of research in future work.

References

[1] N. Allezard, M. Dhome, and F. Jurie. Recognition of 3d textured objects by mixing view-based and model-based representations. In *International Conference on Pattern Recognition*, pages 960–963, Barcelona, Spain, Sep 2000.

[2] Simon Baker, Iain Matthews, Jing Xiao, Ralph Gross, Takeo Kanade, and Takahiro Ishikawa. Real-time non-rigid driver head tracking for driver mental state estimation. In *11th World Congress on Intelligent Transportation Systems*, October 2004.



(a) (b)

Figure 7: Newspaper page detection under occlusion or deformation. (a) Original images. (b) Images with overlaid validation texture.

Figure 8: Superimposing an appropriately deformed CVPR logo on the ICCV T-shirt of Fig 1. The corresponding video sequence is submitted as supplementary material.

[3] A. Bartoli and A. Zisserman. Direct Estimation of Non-Rigid Registration. In *British Machine Vision Conference*, Kingston, UK, September 2004.

[4] A. Baumberg. Reliable feature matching across widely separated views. In *Conference on Computer Vision and Pattern Recognition*, pages 774–781, 2000.

[5] S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition Using Shape Contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(24):509–522, apr 2002.

[6] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. *Comput. Vis. Image Underst.*, 89(2-3):114–141, 2003.

[7] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active Appearance Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6), jun 2001.

[8] D. DeCarlo and D. Metaxas. Deformable Model-Based Shape and Motion Analysis from Images using Motion Residual Error. In *International Conference on Computer Vision*, pages 113–119, Bombay, India, 1998.

[9] G. Dewaele, F. Devernay, and R. Horaud. Point Trajectories and a Smooth Surface Model. In *European Conference on Computer Vision*, Prague, Czech Republic, May 2004.

[10] V. Ferrari, T. Tuytelaars, and L. Van Gool. Simultaneous Object Recognition and Segmentation by Image Exploration. In *European Conference on Computer Vision*, May 2004.

[11] M.A Fischler and R.C Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications ACM*, 24(6):381–395, 1981.

[12] P. Fua. *RADIUS: Image Understanding for Intelligence Imagery*, chapter Model-Based Optimization: An Approach to Fast, Accurate, and Consistent Site Modeling from Imagery. Morgan Kaufmann, 1997. O. Firschein and T.M. Strat, Eds.

[13] P. Fua and Y. G. Leclerc. Object-Centered Surface Reconstruction: Combining Multi-Image Stereo and Shading. *International Journal of Computer Vision*, 16:35–56, September 1995.

[14] S. Granger and X. Pennec. Multi-scale em-icp: A fast and robust approach for surface registration. In *European Conference on Computer Vision*, pages 418–432, Copenhagen, Denmark, 2002.

[15] N.A. Gumerov, A. Zandifar, R. Duraiswami, and L.S. Davis. Structure of Applicable Surfaces from Single Views. In *European Conference on Computer Vision*, Prague, May 2004.

[16] C.G. Harris and M.J. Stephens. A combined corner and edge detector. In *Fourth Alvey Vision Conference, Manchester*, 1988.

[17] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[18] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active Contour Models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[19] V. Lepetit and P. Fua. Towards Recognizing Feature Points using Classification Trees. Technical Report IC/2004/74, EPFL, 2004.

[20] V. Lepetit, J. Pilet, and P. Fua. Point Matching as a Classification Problem for Fast and Robust Object Pose Estimation. In *Conference on Computer Vision and Pattern Recognition*, Washington, DC, June 2004.

[21] V. Lepetit, L. Vacchetti, D. Thalmann, and P. Fua. Fully Automated and Stable Registration for Augmented Reality Applications. In *International Symposium on Mixed and Augmented Reality*, Tokyo, Japan, September 2003.

[22] D.G. Lowe. Object recognition from local scale-invariant features. In *International Conference on Computer Vision*, pages 1150–1157, 1999.

[23] D.G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 20(2):91–110, 2004.

[24] J. Matas, O. Chum, U. Martin, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *British Machine Vision Conference*, pages 384–393, London, UK, September 2002.

[25] K. Mikolajczik and C. Schmid. A performance evaluation of local descriptors. In *Conference on Computer Vision and Pattern Recognition*, pages 257–263, June 2003.

[26] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *European Conference on Computer Vision*, pages 128–142. Springer, 2002. Copenhagen.

[27] F. Mindru, T. Moons, and L. VanGool. Recognizing color patterns irrespective of viewpoint and illumination. In *Conference on Computer Vision and Pattern Recognition*, pages 368–373, 1999.

[28] C. Schmid and R. Mohr. Local grayvalue invariants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(5):530–534, May 1997.

[29] T. Tuytelaars and L. VanGool. Wide Baseline Stereo Matching based on Local, Affinely Invariant Regions. In *British Machine Vision Conference*, pages 412–422, 2000.

[30] L. Vacchetti, V. Lepetit, and P. Fua. Stable real-time 3d tracking using online and offline information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(10):1385–1391, October 2004.