Local Features, All Grown Up

Andrea Vedaldi Stefano Soatto
University of California at Los Angeles
Los Angeles – CA 90095
{vedaldi,soatto}@cs.ucla.edu

Abstract

We present a technique to adapt the domain of local features through the matching process to augment their discriminative power. We start with local affine features selected and normalized independently in training and test images, and jointly expand their domain as part of the correspondence process, akin to a (non-rigid) registration task that yields a (multi-view) segmentation of the object of interest from clutter, including the detection of occlusions. We show how our growth process can be used to validate putative affine matches, to match a given "template" (an image of an object without clutter) to a cluttered and partially occluded image, and to match two images that contain the same unknown object in different clutter under different occlusions (unsupervised object detection).

1. Introduction

Viewpoint invariant features have proven to be a useful tool in the recognition of objects and categories from images. In order to make them invariant, or insensitive, to nuisance factors of the image formation process, one has to trade off discriminative power [30]. For instance, achieving insensitivity to occlusions calls for local image statistics (local features), while increasing their discriminative power is typically achieved by combining several local features into more or less structured collections (graphical models, constellations, or "bags").

We propose a technique to *adapt* local features in a way that is *tailored to the correspondence process,* in order to *augment* their discriminative power. We start with local affine features selected and normalized independently in training and test images, and expand their domain as part of the correspondence process. Correspondence amounts to a (non-rigid) registration task, and the dilation process yields a multi-view segmentation of the object of interest from clutter, including the detection of occlusions.

This process can be interpreted as the simultaneous registration and segmentation of deformable objects in multiple views starting from an "affine seed." It can also be thought of as an implicit 3-D reconstruction of the scene, which enables the recognition of non-planar objects and the discrimination of objects based on their shape [30].

We formalize the feature growth process as an optimization problem, and introduce efficient algorithms to solve it under three different deformation models. Our growth process can be thought of as a region-based segmentation scheme, but indeed it is quite different since it is *unilateral*, i.e. it requires a characterization of the foreground, but not of the background statistics.

Among the applications of our technique are general object recognition tasks, both *supervised*, i.e. given an uncluttered, unoccluded image ("template") of a (3-D, possibly deformable) object of interest, find it in a cluttered image, and *unsupervised*, i.e. given two or more images all containing the same object under different clutter and occlusions, detect and localize the common object. Note that we concentrate on the recognition of specific objects, rather than object categories, as we do not allow intrinsic variability of the object other than the geometric deformations captured by the model. Our goal is to improve the discriminative power of local representations, so that finer discrimination can be performed: We do not just want to tell a face from a bottle; we want to tell *one* particular bottle from another, say with a scratch on it.

1.1. Related work and our contributions

Our technique builds on local affine invariant features [23, 17]. By dilation and alignment, it increases their discriminative power as part of the matching process rather than directly as part of the representation, and extends their validity to non-planar, non-rigid objects. In principle, nothing prevents us from gathering such enlarged regions into constellations or bags [11, 14, 10, 25] although we will confine ourselves to studying *one* feature in isolation to better test the improvement relative to affine descriptors.

Since we combine region growing with registration, our work is also related to [19, 32], although these authors address the problem of global correspondence in a short base-

line setting. [13] propagates affine matches from an (unoccluded) image of an object (template) and a small set of initial seeds. None of these approaches model both viewpoint-induced deformations and the shape of the extracted regions explicitly.

As the selection of the interest region is not determined by the local image statistics alone, but is determined through the matching process, one can think of our technique as a *motion segmentation* procedure [1, 9, 33]. Finally, since the growth process takes part during the alignment (correspondence) from multiple views, our work relates to [8, 15, 5] and other tracking and long-baseline correspondence techniques, although it differs from them computationally.

In order to keep the implementation efficient, we work in a discrete rather than variational setting, much in the spirit of [3, 27]. To this end, we introduce models of regularized region growth that are flexible and result in very efficient algorithms [29]. As opposed to [34] and similarly to [24], the segmentation is local and uses statistics only inside and in the immediate neighborhood of the region.

1.2. Notation

An image is a function $I_t : \Lambda \to \mathbb{R}$, $t = 1, 2, ...$ defined on a discrete lattice $\Lambda = \{0, ..., M\} \times \{0, ..., N\}$. Whenever the argument x of $I_t(x)$ has fractional coordinates, it is intended that bilinear interpolation is being used.

A *feature* (or "interest region") is specified by a (binary or smooth) *window* function $H(x) : \Lambda \to [0, 1]$ and a *support* $\Omega \subset \Lambda$, a subset of the image domain. Although related to the window $H(x)$, Ω is *not* necessarily its support $\operatorname{supp} H \triangleq \{x : H(x) \neq 0\}$; its precise meaning will be specified in Sect. 2.1. A *feature match* (H_i, w_{ij}) is a window $H_i(x)$ describing the interest region on one image $I_i(x)$, together with a regular warping (diffeomorphism) $w_{ij} : \mathbb{R}^2 \to \mathbb{R}^2$ onto the corresponding region in another image $I_j(x)$.

2. A model of feature growth

Our method starts with a putative feature match $(H, w) \triangleq (H_1, w_{12})$ initialized from an "affine seed" as in [23, 17]. An *affine seed* is a pair of corresponding elliptical regions (Ω_1, Ω_2) in $I_1(x)$ and $I_2(x)$ respectively. The regions are related by an affine warp $w(x) = Ax + T$, $(A, T) \in \mathbb{GL}(2) \times \mathbb{R}^2$ which is fixed (by $w\Omega_1 = \Omega_2$) up to a rotation $R(\theta) \in \mathbb{SO}(2)$. We estimate the residual rotation by maximizing the normalized cross-correlation (NCC) of the appearance of the two regions. The region Ω_1 and the transformation (A, T) are then used to initialize the feature match (H, w).

We grow the initial match by trading off dilation of the window $H(x)$ and quality of the alignment w, expressed by the following cost functional:

$$E(H, w, \mu) = \sum_{x \in \Lambda} H(x)[(\mu(I_1) - I_2 \circ w)(x)]^2 - \alpha \left(\sum_{x \in \Lambda} H(x) - \beta \mathcal{R}(H) \right) + \gamma \mathcal{Q}(w). \quad (1)$$

The first term is a sum of squared difference (SSD) representing the quality of the local alignment. Minimizing this term has two effects: to select the warp w that best aligns the interest regions and, of course, to shrink the region (the global minimizer of this term is $H(x) = 0 \; \forall \; x$). Minimizing the second term favors instead larger regions. There is a simple interpretation of the control parameter $\alpha \in \mathbb{R}_+$: $\sum H(x)$ can be thought as the area of the region[1] and α as the mean squared residual that we are willing to absorb within the region. The terms $\mathcal{R}(H)$ (Sect. 2.1) and $\mathcal{Q}(w)$ (Sect. 2.2) are regularization terms for the region H and warp w respectively. The function $\mu : \mathbb{R}_+^\Lambda \to \mathbb{R}_+^\Lambda$ is a pre-processing operator that can be used to compensate for other factors affecting the range of the image, such as illumination (Sect. 2.3). The goal is to find H, w and μ by alternating minimization of E, which we discuss in the following sections.

2.1. Region model

The region growth is determined by a controlled evolution of the window function $H(x)$. There are several possible choices for the model of the window ranging from simple parametric models that allow only a limited set of shapes (ellipses, rectangles, etc.) to non-parametric models that enable regions with arbitrary shape (up to topological and smoothness constraints). In order to explore this spectrum of options, we experimented with three models, explained next. Since in this section we focus on the region only, we rewrite the cost (1) as

$$E(H) = \sum_{x \in \Lambda} H(x)(D(x)^2 - \alpha) - \alpha\beta\mathcal{R}(H) + \text{const.} \quad (2)$$

where we have defined the residual $D(x) \triangleq \mu(I_1)(x) - (I_2 \circ w)(x)$.

Elliptical region. The first model, fully parametric, is a smoothed elliptical window

$$H(x; p) \triangleq \phi(y^\top y), \quad y = A(p)^{-1}(x - T(p)), \quad x \in \mathbb{R}^2$$

where $\phi \in C^\infty(\mathbb{R}_+ \to [0, 1])$ is a non-increasing function such that $\phi(0) = 1$ and $\phi(+\infty) = 0$ and $(A(p), T(p)) \in \mathbb{GL}(2) \times \mathbb{R}^2$ is the affine map that brings the unit circle onto the elliptical region. The window is parametrized by

[1] This is exactly the case for binary regions when $\beta = 0$.

Algorithm 1 Growing binary free-form regions

1: Pre-compute dilation cost table $\mathrm{Lkp} : \{0,1\}^8 \to \mathbb{R}$.
2: Make heap of $\{(D^2(x_+) - \alpha(1 - \beta\,\mathrm{Lkp}(\mathcal{N}_\Omega(x_+))), x_+), x_+ \in \partial_+\Omega\}$
3: **loop**
4: Pop minimal element (c_+, x_+) from the heap. Stop if $c_+ > 0$.
5: $\Omega \leftarrow \Omega \cup \{x_+\}$ and $H \leftarrow \chi_{\Omega_+}$.
6: Add missing 4-neighbors of x_+ to the heap.
7: Update the cost of the 4-neighbors of x_+ in the heap.
8: **end loop**

the vector $p \in \mathbb{R}^6$ as $\mathrm{vec}\, A = (p_1, p_2, p_3, p_4)^\top$ and $T = (p_5, p_6)^\top$, where vec denotes the stacking operator. This model does not make explicit use of the feature support Ω; it is however handy to define it as the nominal support of the window $\Omega = \{x : H(x) > \tau\}$, for some small value of τ (e.g. $\tau = 1\%$).

Since this model is fully constrained, the regularization term $\mathcal{R}(H)$ in eq. (2) is unnecessary. The resulting minimization problem can be solved by Gauss-Newton (GN) or any other descent technique. In the experiments we combined steepest descent (SD) with GN for reliable and fast convergence.

Binary free-form region. A binary free-form region is the characteristic function $H(x) = \chi_\Omega(x)$ of a domain $\Omega \subset \Lambda$ that has 4-neighbors connectivity. The regularization term $\mathcal{R}(H)$ is the length of the 8-ways discrete perimeter $\pi_8(\Omega)$ of the set Ω. The representation allows for changes in topology, even if these are discouraged by the regularization (too many "holes" will increase the length of the perimeter). The cost functional (2) assumes the form

$$E(H) = \sum_{x \in \Lambda} H(x)(D(x)^2 - \alpha) + \alpha\beta\pi_8(\Omega) + \text{const.}$$

To maximize of $E(H)$ we add to Ω the pixel x_+ belonging to the outer border $\partial_+\Omega$ (dilation) or we remove the pixel x_- belonging to the inner border $\partial_-\Omega$ (contraction) that most decreases the cost function (SD). Here we discuss only dilation moves, as contraction moves are similar. The updated window $H_+ = \chi_{\Omega \cup \{x_+\}}$ has cost

$$E(H_+) = E(H) + (D(x_+)^2 - \alpha) + \alpha\beta\left(\pi_8(\Omega \cup \{x_+\}) - \pi_8(\Omega)\right). \quad (3)$$

The term $\pi_8(\Omega \cup \{x_+\}) - \pi_8(\Omega)$ is very efficient to compute. In fact, it depends only on the tuple $\mathcal{N}_\Omega(x_+) \triangleq (\chi_\Omega(x) : x \text{ is 8-neighbor of } x_+)$ and can be pre-computed and stored in a lookup table of just 256 entries, leading to Algorithm 1. This algorithm is similar to [29] and reminiscent of the discrete level-sets of [27].

Algorithm 2 Growing smooth free-form regions

1: $D^2_\sigma \leftarrow g_\sigma * D^2$
2: $H \leftarrow g_\sigma * \chi_\Omega$
3: Make heap of $\{(D^2_\sigma(x_+) - \alpha(2H(x_+) + g_0), x_+), x_+ \in \partial_+\Omega\}$
4: **loop**
5: Pop minimal element (c_+, x_+) from the heap. Stop if $c_+ > 0$.
6: $\Omega \leftarrow \Omega \cup \{x_+\}$ and $H \leftarrow H + g_\sigma * \delta_{x_+}$.
7: Add missing 4-neighbors of x_+ to the heap
8: Update the cost of neighbors within the support of the kernel g_σ in the heap.
9: **end loop**

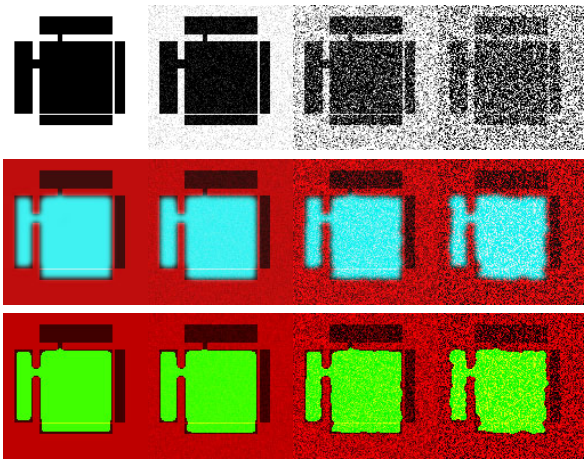
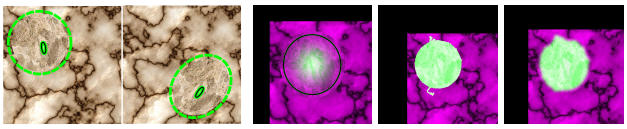


Figure 1. *Smooth free-from region.* We tested Algorithm 2 on the test images shown on the top. We show in the middle row (cyan) the smooth window $H(x)$ and in the bottom row (green) the support Ω. The parameter σ has been chosen so that the region can squeeze through the left corridor, but not the upper corridor, and past the bottom line, but not the right line. The computation requires a fraction of a second and the result is consistent even if a large amount of noise is injected. Note the regularizing effect of the kernel g_σ: the boundary of Ω is fairly smooth despite the fact that it is grown by discrete steps (one pixel per time).

This model has two drawbacks: the window $H(x)$ is not smooth and the amount of regularization that can be imposed on the shape of the region is limited by the discrete nature of the steps that are used in the descent (too much regularization can block growth). To overcome these restrictions we turn to the smooth free-form region model described next.

Smooth free-form region. A smooth free-form region is obtained by smoothing a binary free-form region. The window $H(x)$ is given by the convolution $(g_\sigma * \chi_\Omega)(x)$, $x \in \Lambda$ where g_σ is a Gaussian kernel of standard deviation σ. This yields a smooth window *and* a very efficient regularization



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

Figure 2. *Effect of the region model in capturing correspondence between regions.* (a) A round region with different texture from the background (left) is moved and deformed affinely on the right image. Within these regions an affine seed is detected by Harris-Affine (in green). Note that the regions do not have well-defined (intensity) boundaries. The goal is to extend the seed to capture the elliptical region based on two-view correspondence. This can be thought of as a "stereoscopic texture segmentation" or as a "motion segmentation" procedure [9, 33, 1]. The effects of the choice of region are shown in (b)-(d). In (b) the region is by construction elliptical, and its domain captures the texture boundary. In (c) the region is free-form, but captures the elliptical shape, modulo some sprouts outside the region where the background happens to match in the two views; in (d) the sprouts are contained by the smooth free-form region.

criterion, as explained next. In eq. (2) we set $\beta = 1$ and $\mathcal{R}(H) = \sum_{\Lambda/\Omega} H(x)$ so that

$$E(H) = \sum_{x\in\Lambda} H(x)D(x)^2 - \alpha \sum_{x\in\Omega} H(x) + \text{const.}$$

Since $H = g_\sigma * \chi_\Omega$ is small where the boundary of the region has high curvature or where the region is thin, the regularization favors compact and smooth regions. Like for binary regions, minimization of $E(H)$ is fast. Again we discuss only dilation moves. Given the window $H = g_\sigma * \chi_\Omega$, $\Omega \subset \Lambda$, we need to find the pixel $x_+ \in \partial_+\Omega$ for which the new window $H_+ = g_\sigma * \chi_{\Omega\cup\{x_+\}}$ has the lowest possible cost $E(H_+)$. We have

$$\begin{aligned} E(H_+) &= \sum_{x\in\Lambda} D(x)^2 \left(g_\sigma * (\chi_\Omega + \delta_{x_+})\right)(x) \\ &\quad - \alpha \sum_{x\in\Omega} \left(g_\sigma * (\chi_\Omega + \delta_{x_+})\right)(x) \\ &= E(H) + (g_\sigma * D^2)(x_+) - \alpha(2H(x_+) + g_0) \end{aligned} \quad (4)$$

where $\delta_{x_+} = \delta(x - x_+)$ is the Kronecker's delta and $g_0 \triangleq g_\sigma(0)$. The map $g_\sigma * D^2$ can be conveniently pre-computed, leading to Algorithm 2. Figure 1 shows some examples of regions grown using Algorithm 2; Figure 2 compares the three models as they grow the same affine match.

2.2. Warping model

The deformation induced on the image domain by changes in viewpoint can be rather complex, depending on the shape of the scene [30]. In particular, occlusions cause such a transformation to be globally non-invertible, and there is no way to distinguish a-priori an occlusion (a portion of the scene disappearing under another) from a "collapse" (a portion of the scene being warped onto a subset of measure zero). Therefore, we have to impose restrictions on the local structure of the warping w (or equivalently on the motion and curvature of the underlying 3-D shape), for instance that it be locally continuous and bijective and, for reasons of computational efficiency, finitely parametrized.

We have experimented with three classes of transformations: affine, homography (corresponding to locally planar regions), and thin-plate spline. The last model is well suited to non-planar or deforming (non-rigid) scenes, but it is in general not globally invertible (thin-plate splines can fold).[2] We optimize the functional (1) using Gauss-Newton as in [3]. The derivation of the GN algorithm for these models is standard.

Affine warp and homography. The affine warp and the homography are finite dimensional and they do not need to be regularized, so that $\gamma = 0$ in (1).

Thin-plate spline. The thin-plate spline warp is given by [6]

$$w(x) = \begin{bmatrix} T & A & W \end{bmatrix} \begin{bmatrix} 1 \\ x \\ U(\|x - y^{(:)}\|) \end{bmatrix}$$

where (A, T) is an affine transformation, $W \in \mathbb{R}^{2\times K}$ is a matrix of weights, $y^{(:)} = (y^{(1)}, ..., y^{(K)})$ denotes collectively the K control points $y^{(k)} \in \mathbb{R}^2$ and $U(\|x - y^{(:)}\|) = \left[\|x - y^{(:)}\|^2 \log \|x - y^{(:)}\|^2\right]$ is the matrix of the radial basis functions of the spline. The matrices T, A and W are uniquely determined by the transformed control points $\bar{Y} = \left[w(y^{(1)}) \quad ... \quad w(y^{(K)})\right]$, yielding a relation

$$w(x; \bar{Y}) = \begin{bmatrix} \bar{Y} & 0 \end{bmatrix} \phi(x; y^{(:)}), \quad \phi(x; y^{(:)}) \in \mathbb{R}^{K+3} \quad (5)$$

which is *linear* in the parameters $\bar{Y}$.

Regularization is controlled both by the number of points K and the *stiffness* (bending energy)

$$\mathcal{Q}(\bar{Y}) = \frac{\gamma}{2}(e_1 \otimes e_1 + e_2 \otimes e_2)^\top \operatorname{vec}(\bar{Y} S \bar{Y}^\top)$$

where $\otimes$ denotes the Kronecker's product, $S \in \mathbb{R}^{2\times 2}$ is the *stiffness matrix* and (e_1, e_2) is the standard basis of $\mathbb{R}^2$.

Optimization can be performed with GN,[3] but this is

[2] An interesting approach to this problem would be to regularize the warps based on priors on the shape [4, 26, 31]. This, however, is beyond the scope of this paper.

[3] As noted in [20], the linearity of (5) makes the estimation of the gradient efficient. In order to write the equations for the GN iteration, one also needs the gradient and the Hessian of the stiffness term, which are

$$\frac{\partial \mathcal{Q}(q)}{\partial q^\top} = \sum_{i=1}^{2} e_i^\top \bar{Y} S \otimes e_i^\top, \quad \frac{\partial^2 \mathcal{Q}(q)}{\partial q^\top \partial q} = S \otimes I_2$$

where $q \triangleq \operatorname{vec} \bar{Y}$ and $I_2 \in \mathbb{R}^{2\times 2}$ is the identity matrix (see [18] for more details on the notation).

quite costly as each control point has a global influence on the warp. We drastically accelerate the computation by approximating the TPS by a piecewise-affine warp (PWA, [2]) by imposing on its vertices the same regularization (stiffness) of the TPS. The PWA is intrinsically more efficient because each control point has an effect limited to a few triangles of the mesh. We also make use of the inverse compositional algorithm [2] in place of GN, which is much faster.

2.3. Matching criterion

Appearance matching in model (1) uses a simple sum of squared difference criterion. The adjustment function $\mu = (\mu_1, \mu_2)$ is an affine scaling $\mu(I_1) = \mu_1 I_1 + \mu_2$ that accounts for global changes in the illumination. As such, μ can be determined in closed form given H and w; alternatively, its optimization can be combined in the GN iteration for w. Coarse illumination factors can be eliminated in other ways. For instance, in some of the experiments we normalize the images via

$$\mu(I) = \frac{I - g_\sigma * I}{\sqrt{g_\sigma * I^2 - (g_\sigma * I)^2}} \tag{6}$$

where g_σ denotes an isotropic Gaussian kernel of variance $\sigma^2 I_2$, with I_2 the 2×2 identity matrix. Note, however, that this operator has to be applied to both I_1 and $I_2 \circ w$, which makes the optimization more complex. For further comments on the matching criteria, see Sect. 4.

3. Experiments

Global projective registration from a seed. The first simple experiment illustrates how a *single* local feature can grow to encompass the entire image. The two images of Fig. 3 are related by an homography; their registration yields a projective "mosaic" which can be obtained efficiently by matching a single feature and then growing it to capture the entire overlapping domain.

Growing increases discriminative power. The second experiment correlates the growth rate of the features to their initial overlap. In [22] the quality of an affine seed (Sect. 2) is evaluated by means of the *overlap error*

$$\epsilon \triangleq 1 - \frac{|\Omega_1 \cap \overline{w}^{-1}\Omega_2|}{|\Omega_1 \cup \overline{w}^{-1}\Omega_2|} \tag{7}$$

where $\overline{w}$ is the ground truth viewpoint deformation. We used the "viewpoint change" dataset of [22] and their code in order to extract Harris-Affine regions and compute ϵ.

We ran the algorithm on several affine seeds using elliptical regions and homographies. As in the dataset there are almost no occlusions, correct seeds (overlap error less than 100%) should grow indefinitely and incorrect seeds (100% overlap error) should not grow at all. To check wether this is the case, in Fig. 4 we plot the average dilation ratio of the matches (left) and the probability of each match of being dilated (right) as a function of the initial overlap error. As desired, the algorithm grows quickly correct matches with up to 50% of initial overlap error and does not grow almost any of the incorrect matches. The algorithm does not perform equally well for correct seeds that have initial error exceeding 50%. This is because we focus on discriminating correct versus incorrect matches rather than trying to fix seeds of poor quality. If this is desired, a robust initialization step can be added (for instance as described in [13]).

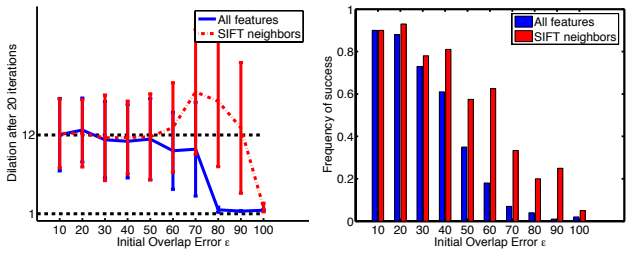


Figure 4. *Growing increases discriminative power.* On the left we show the average dilation ratio of the features as a function of the initial overlap error (7). As we stopped the algorithm after 20 iterations, the graph gives also an idea of the speed of the dilation. This statistic includes only features with a ratio ≥ 1. On the right we show how frequently this ratio is in fact bigger than 1 (dilation), again as a function of the initial overlap error. The experiment is repeated for all affine seeds and for affine seeds that are SIFT [21] neighbors. See text for further details.

As our final goal is to discriminate features beyond the power of their descriptors, we repeated this experiment for those affine seeds that are also SIFT neighbors.[4] The performance of the algorithm does not deteriorate; in particular, almost all matches determined incorrectly by SIFT are invalidated by our criterion, while correct matches are preserved. As a side effect, the algorithm is also more robust to poor initial overlap, probably because seeds which are SIFT neighbors have, if not good geometric correspondence, at least similar appearance.

Finding a known object in clutter. This experiment tests the capability of our method to find – in clutter – an object for which an uncluttered image is given as a training set (or "template"). It is similar in spirit to the experiments of [13, 12] and many other object recognition systems [21]. Since we use a deformable object (the Garfield book in Fig. 5), we use the thin-plate spline warp and the smooth free-form region model. The figure shows the detection/segmentation results, together with the alignment to the template and the estimated deformation. Note that the latter two quantities are meaningful only locally to the segmented area (so it is not a problem if the template does not align well outside

[4] More precisely: for each region Ω_1 of image $I_1(x)$ we selected the three closest regions Ω_2 of image $I_2(x)$ in terms of SIFT distance.

(a) Initial affine match (b) Grown window (c) Overlap (d) Residual

Figure 3. *Growing a mosaic from one feature.* A single feature detected and matched on two images related by an homography grows to capture the entire overlapping domain, yielding a projective mosaic. Here the region model is elliptical and the warp is an homography. (a) Initial affine matches (very small, so we inlay a magnified version) (b) interest region Ω (green ellipse) and window $H(x)$ (green shading) (c) overlap between the two images (perfect overlap results in unsaturated color) (d) residual.

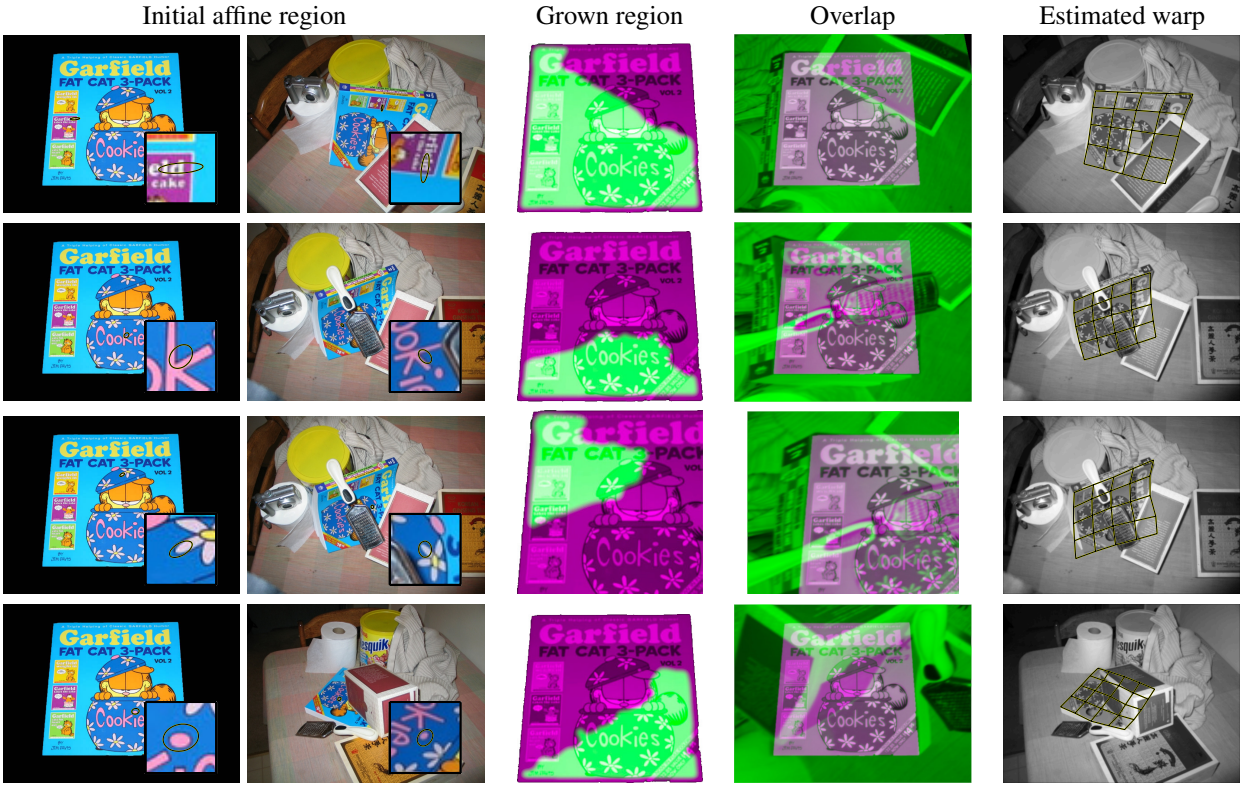


Figure 5. *Object detection in clutter.* The left column shows various training/test image pairs. Each pair shows the initial affine match that is grown by the algorithm. Since the object is non-rigid, we used the thin-plate spline model for the warp and the smooth free-form model for the region. In a few cases a portion of the visible area is not included in the region: This is due to the non-uniform illumination (difficult to see with the naked eye but quantitatively significant) which is not compensated by the global model (1). It would not be difficult to extend μ to account for more general contrast functions [7] (Sect. 2.3 and 4).

that area.)

Detecting a common object in cluttered scenes. While the previous experiment can be thought of as "supervised" detection, since a template of the object is given, here we address "unsupervised" detection, that is the problem of detecting the common portion of two images, without a pre-segmented sample of the region of interest. The data are four images that share a single object (a bottle). In Fig. 6 we show that the algorithm is generally capable of segmenting the common object from clutter (see the caption for more details).

4. Discussion

We have proposed a method that increases the information content of local features by maximizing their support. We have shown that the growth rate can be used to validate putative affine matches; the criterion is especially useful to verify matches that have been hypothesized on the basis of the distance between local descriptors. We have seen that the dilated support delineates segments of both known and

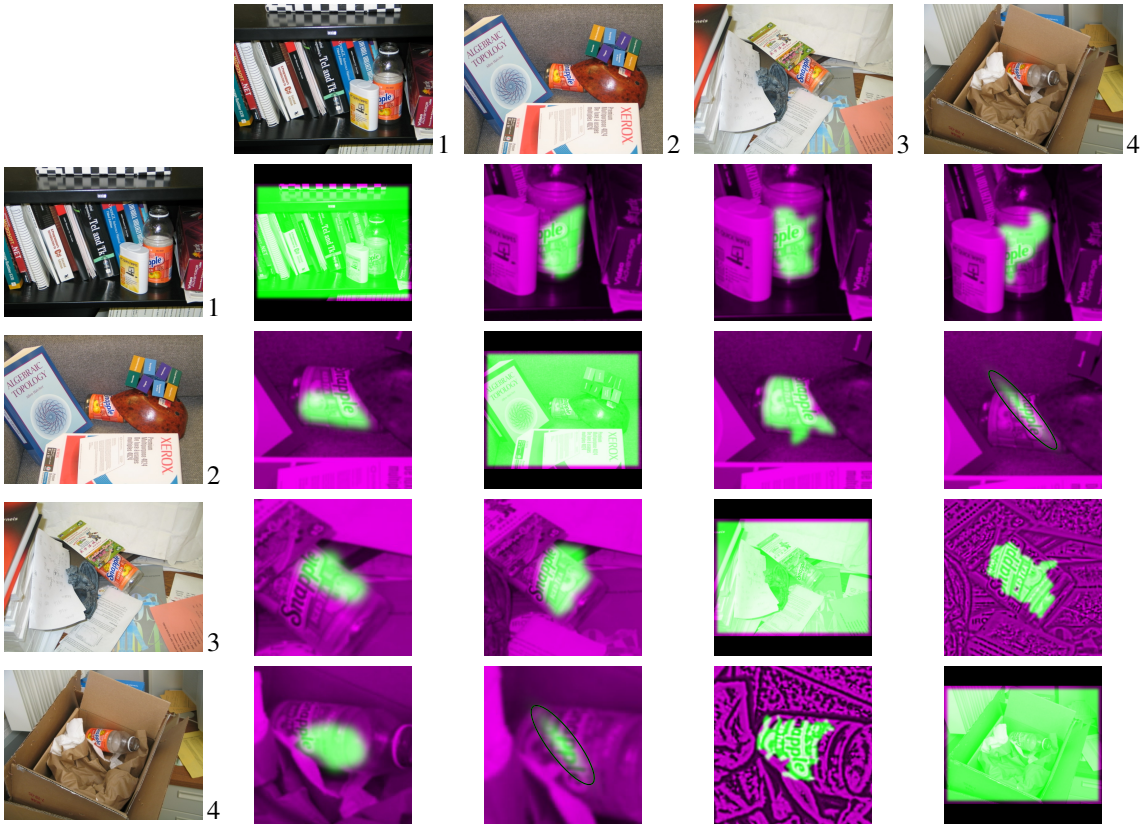


Figure 6. *Unsupervised detection in clutter.* We show a series of four images (1-4) portraying the same object (a bottle) in different clutter. The algorithm is tested on each image pair (in both directions) in order to detect the common object. The best SIFT matches of Harris-Affine features on the bottle are expanded and the best result for each pair is kept. Along the diagonal, as we test identical pairs, there is no deformation and the region extends to encompass the whole image domain. In the pairs $(4, 3)$ and $(3, 4)$ the complex reflectance and the particular image deformation (which has high stiffness) requires preprocessing according to eq. (6) in order to enable matching. In the pairs $(2, 4)$ and $(4, 2)$ part of the background is almost identical (once color is removed). Therefore, the SSD criterion cannot discriminate between the bottle and the background. One can overcome this ambiguity by using a more constrained region model (elliptical) at the cost of reducing the segmented area. A more principled solution is indicated in Sect. 4.

unseen objects from images with clutter. The latter task is significantly more complex since no uncluttered, unoccluded view of the object of interest is ever available.

Unsupervised detection in clutter is complicated by the fact that certain portions of the background might match accidentally, which is especially easy if the background is uniform. This problem can be properly addressed by ensuring that the region grows where matching is *non-accidental*, that is in areas of the two images that have an appearance which is at the same similar *and* contains "enough structure" [17, 28]. While this constraint can be imposed as a regularization term on the region, a better solution is to substitute the SSD matching criterion with one that incorporates directly this requirement (see for example [16]). Thus the issue is more computational than theoretical, as these measures are significantly more expensive to optimize than SSD.

Acknowledgments. Research sponsored by ONR N00014-03-1-0850:P0001 and AFOSR F49620-03-1-0095.

References

[1] M. Allan, M. K. Titsias, and C. K. I. Williams. Fast learning of sprites using invariant features. In *Proc. BMVC*, 2005.

[2] S. Baker and I. Matthews. Equivalence and efficiency of image aligment algorithms. In *Proc. CVPR*, 2001.

[3] S. Baker and I. Matthews. Lucas-Kanade 20 years on: A unifying framework: Part 1. Technical Report CMU-RI-TR-02-16, CMU, 2002.

[4] M. J. Black, Y. Yacoob, A. D. Jepson, and D. J. Fleet. Learning parametrized models of image motion. In *Proc. CVPR*, 1997.

[5] A. Blake. Visual tracking: a research roadmap. In O. Faugerasn, Y. Chen, and N. Paragios, editors, *Mathematical Models of Computer Vision: The Handbook*. Springer, 2005.

[6] F. L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *PAMI*, 11(6):567–585, 1989.

[7] V. Caselles, B. Coll, and J.-M. Morel. Topographic maps and local contrast changes in natural images. *IJCV*, 33(1), 1999.

[8] T. F. Cootes, S. Marsland, C. J. Twining, C. J. Taylor, and K. Smith. Groupwise diffeomorphic non-rigid registration for automatic model building. In *Proc. ECCV*, 2004.

[9] D. Cremers and S. Soatto. Motion competition: A variational approach to piecewise parametric motion segmentation. *IJCV*, 2004.

[10] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *IEEE CVPR Workshop of Generative Model Based Vision*, 2004.

[11] P. F. Felzenszwalb and D. P. Huttenlocher. Pictorial structures fod object recognition. *IJCV*, 61(1), 2005.

[12] R. Fergus, P. Perona, and A. Zisserman. A sparse object category model for efficient learning and exhaustive recognition. In *Proc. CVPR*, 2005.

[13] V. Ferrari, T. Tuytelaars, and L. Van Gool. Simultaneous object recognition and segmentation from single or multiple model views. *IJCV*, 2005.

[14] D. A. Forsyth, J. Haddon, and S. Ioffe. Finding objects by grouping primitives. In D. A. Forsyth, J. L. Mundy, V. D. Gesù, and R. Cipolla, editors, *Shape, contour and grouping in computer vision*. Springer-Verlag, 2000.

[15] G. D. Hager and P. N. Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *PAMI*, 1998.

[16] G. Hermosillo and O. Faugeras. Dense image matching with global and local statistical criteria: A variational approach. In *Proc. CVPR*, 2001.

[17] T. Kadir and M. Brady. Scale saliency: A novel approach to salient feature and scale selection. In *International Conference Visual Information Engineering*, 2003.

[18] D. B. Kinghorn. Integrals and derivatives for correlated gaussian fuctions using matrix differential calculus. *International Journal of Quantum Chemestry*, 57:141–155, 1996.

[19] M. Lhuillier. Efficient dense matching for textured scenes using region growing. In *Proc. ECCV*, 1998.

[20] J. Lim and M.-H. Yang. A direct method for modeling non-rigid motion with thin plate spline. In *Proc. CVPR*, 2005.

[21] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2(60):91–110, 2004.

[22] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. In *Proc. CVPR*, 2003.

[23] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *IJCV*, 1(60):63–86, 2004.

[24] E. Pichon, A. Tannenbaum, and R. Kikinis. A statistically based flow for image segmentation. *Medical Image Analysis*, 2004.

[25] J. Ponce, S. Lazebnik, F. Rothganger, and C. Schmidt. Toward true 3D object recognition. In *Reconnaisance de Formes et Intelligence Artificelle*, 2004.

[26] M. Salzmann, S. Ilic, and P. Fua. Physically valid shape parameterization for monocular 3-D deformable surface tracking. In *Proc. BMVC*, 2005.

[27] Y. Shi and W. C. Karl. A fast level set method without solving PDEs. In *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, 2005.

[28] B. Triggs. Detecting keypoints with stable position, orientation, and scale under illumination changes. In *Proc. ECCV*, 2004.

[29] J. N. Tsitsiklis. Efficient algorithms for globally optimal trajectories. *IEEE Trans. on Automatic Control*, 40(9), 1995.

[30] A. Vedaldi and S. Soatto. Features for recognition: Viewpoint invariance for non-planar scenes. In *Proc. ICCV*, 2005.

[31] A. Vedaldi and S. Soatto. Viewpoint induced deformation statistics and the design of viewpoint invariant features: Singularities and occlusions. In *Proc. ECCV*, 2006.

[32] Y. Wei and L. Quan. Region-based progressive stereo matching. In *Proc. CVPR*, 2004.

[33] J. Wills, S. Agarwal, and S. Belongie. What went where. In *Proc. CVPR*, 2003.

[34] S.-C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *PAMI*, 18(9):884–900, 1996.