Exploiting Part-Based Models and Edge Boundaries for Object Detection

Josephine Sullivan  Oscar Danielsson  Stefan Carlsson
Royal Institute of Technology, Stockholm, Sweden.
E-mail: {sullivan, osda02, stefanc}@csc.kth.se

Abstract

This paper explores how to exploit shape information to perform object class recognition. We use a sparse part-based model to describe object categories defined by shape. The sparseness allows the relative spatial relationship between parts to be described simply. It is possible, with this model, to highlight potential locations of the object and its parts in novel images. Subsequently these areas are examined by a more flexible shape model that measures if the image data provides evidence of the existence of boundary/connecting curves between connected hypothesized parts. From these measurements it is possible to construct a very simple cost function which indicates the presence or absence of the object class.

The part-based model is designed to decouple variations due to affine warps and other forms of shape deformations. The latter are modeled probabilistically using conditional probability distributions which describe the linear dependencies between the location of a part and a subset of the other parts. These conditional distributions can then be exploited to search efficiently for the instances of the part model in novel images. Results are reported on experiments performed on the ETHZ shape classes database that features heavily cluttered images and large variations in scale.

1. Introduction

Different object classes are distinctive because of characteristics of certain attributes. Not all the same attributes are used for all object classes. For instance, fruits are visually defined primarily by their colour and shape, animals by a combination of texture and shape and some man made objects by the presence of distinctive parts in a very rigid spatial arrangement such as the wheels for cars and bicycles. It would seem though that shape is the most commonly included attribute. Thus a key component of any successful object recognition algorithm must be the robust integration of shape based information. Indeed recent activity in computer vision research has reflected this notion. Impressive results have been achieved in object recognition and localization by describing shape as the relative spatial arrangement of parts and contour fragments [10, 18, 17, 8], as a deformation of a dense set of points, typically corresponding to the boundary of an object [3, 2, 9, 11] or implicitly [14]. Each of these have influenced the direction of this work.

This paper continues this trend by exploring the issue of exploiting shape information to perform object class recognition. However, there is an inherent difficulty in finding shapes that deform, even if somewhat systematically, in clutter. If one has a rigid shape model then one is robust to the clutter but, can only match to a very limited number of very specific shapes with that model. While on the other hand allowing a more flexible shape model introduces a lack of robustness to clutter. Thus the challenge is to define a shape model that allows some flexibility in the set of shapes it can be matched to but not at the cost of matching to anything and everything when the scene is cluttered. We try to achieve this with a two stage process that involves both a part-based fairly rigid model with respect to shape (minus affine transformations) and then a more shape flexible process that looks for boundary curves or the connectedness between appropriate pairs of parts. The latter is used to confirm the presence of the object at the location highlighted by the part-based model and also produces an estimate of the boundary curves of the object.

At the first stage of the process we consider an object as being composed of parts. These parts can potentially be defined at varying resolutions and degrees of sophistication from oriented edgels to whole objects. Ideally though, we want to be able to handle parts that represent common small boundary and edge structure. This would at a future date facilitate the sharing of parts among object classes. However, such parts occur frequently in images. Therefore any part-based model exploiting the relative spatial arrangement of the parts has to cope with a potentially large amount of clutter detections. Another issue is that, as of yet there is no robust generic mathematical definition of shape that exclusively captures the variations in an object class’s shape. However, intuitively it seems that modeling the spatial relations for a separated sparse set of parts is an easier task and also less sensitive to intra-class changes than for a dense set.
We envisage using potentially from 5 to 12 parts to describe an object. The main novelty in the paper is how we construct a joint probability distribution to describe the shape variation of our parts decoupled from affine variations and in terms of simple conditional probability distributions. The latter can also be used to efficiently search for instances of the part model in a novel image, as shown in Section 2.

The approach for describing the parametric model of the boundary curves and their detection is presented in Section 3. While after discussion about implementation issues, results are shown of applying the method to object classes that are mainly defined by their shape in section 5.

2. A part-based model of shape

First we introduce some notation. The object consists of $P$ parts. The object’s configuration, $X'$, corresponds to the 2d position of its parts in the image, that is $X' = \{x_1, \ldots, x_P\}$ with each $x_k \in \mathbb{R}^2$. Given a novel configuration $X'$ the challenge is to decide if it was generated by an instance of the object class we are considering, see figure 1. This is typically achieved by comparing $X'$ either directly to configurations known to have been generated by examples from the object class or to a model of shape learned from such configurations. Ideally the comparison is performed in a manner invariant to common transforms which do not affect the shape such as translation and scaling. In the following we propose an approach based on learning a model of the variation in a object’s configuration which attempts to decouple affine variations and shape variation.

2.1. Specification - exploiting relative spatial dependency

Assume there is a probability distribution $p(x_1, x_2, \ldots, x_P)$ that has non-zero values for configurations from an object class and zero otherwise. It can be decomposed trivially by repeatedly using the product rule:

$$p(x_1, \ldots, x_P) = p(x_{\sigma(1)}) \prod_{j=2}^{P} p(x_{\sigma(j)}|x_{\sigma(1)}, \ldots, x_{\sigma(j-1)})$$

where $\sigma : \{1, \ldots, P\} \rightarrow \{1, \ldots, P\}$ is a permutation defining the order of the decomposition. Without making any simplifying assumptions, specifying and learning all of the conditional distributions is no easier than learning and specifying the joint distribution $p(x_1, x_2, \ldots, x_P)$. Frequently, simplifications based on assuming conditional independence between parts are made, for example the star models of [10, 8]. However, somewhat complementarily we advocate an approach where the dependencies between the parts’ locations are exploited. Clearly, for many objects there are parts whose location can be accurately predicted if the location of a subset of the parts has been given. A basic way to model this is with a linear predictor.

Therefore, we hypothesize that there exists a permutation $\sigma$ and a vector $\beta = (\beta_1, \ldots, \beta_{j-1})$ such that

$$x_{\sigma(j)} = \sum_{k=1}^{j-1} \beta_k x_{\sigma(k)} + \alpha + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \Sigma)$$

with $\sum_{k=1}^{j-1} \beta_k = 1$, $\alpha \in \mathbb{R}^2$ and $\epsilon$ is Normally distributed. If this model is valid then $p(x_{\sigma(j)}|x_{\sigma(1)}, \ldots, x_{\sigma(j-1)})$ equals

$$p \left( x_{\sigma(j)} \Big| \sum_{k=1}^{j-1} \beta_k x_{\sigma(k)} + \alpha, \Sigma \right)$$

2.1.1 Partial affine invariance

This formulation of the conditional probability distributions in equation (2) besides simplicity has the following additional benefit. If the parts undergo a global affine transformation of a linear transformation $A$ and a translation $b$:

$$x'_j = Ax_j + b \quad \text{for } j = 1, \ldots, P$$

then it is easy to show that $p(x'_{\sigma(j)}|x'_{\sigma(1)}, \ldots, x'_{\sigma(j-1)})$ is

$$\mathcal{N} \left( \sum_{k=1}^{j-1} \beta_k x'_{\sigma(k)} + A\alpha, A\Sigma A^T \right)$$

Thus the same $\beta_k$’s describe the linear dependencies between parts for all their affine transformations. This property has been exploited before in the computer vision community and is known to as affine shape [19, 4]. However, clearly even if $\alpha = 0$ the set-up is not completely affine invariant as the covariance in equation (4) is determined by the affine warp from the reference frame in which the original $\Sigma$ is learned to the current frame. In general this affine warp will not be known. Therefore

$$p(x'_1, \ldots, x'_P) = \int_A p(x'_1, \ldots, x'_P|A) p(A) \, dA$$

This can be approximated by

$$p(\hat{A}) p(x'_{\sigma(1)}, x'_{\sigma(2)}, x'_{\sigma(3)}|\hat{A}) \prod_{j=4}^{P} p(x'_{\sigma(j)}|x'_{\sigma(1)}, \ldots, x'_{\sigma(j-1)}, \hat{A})$$

if $\hat{A}$ is an estimate of the linear transformation for which the above integrand achieves a maximum. For practical purposes though $\hat{A}$ is estimated from correspondences $x'_{\sigma(k)} \mapsto x_{\sigma(k)}$, $k = 1, \ldots, j - 1$. The minimum number of correspondences required to compute $\hat{A}$ is three, hence equation (6). Now we define a spatially plausible configuration as one for which $p(x'_1, x'_2, \ldots, x'_P) > 0$ and this only occurs if and only if

$$p(\hat{A}) p(x'_{\sigma(1)}, x'_{\sigma(2)}, x'_{\sigma(3)}|\hat{A}) > 0 \quad \text{and} \quad p(x'_{\sigma(j)}|x'_{\sigma(1)}, \ldots, x'_{\sigma(j-1)}, \hat{A}) > 0$$

(7)
for \( j = 4, \ldots, P \). The first condition imposes that the estimate of \( \hat{A} \) calculated from the initial three correspondences is consistent with prior distribution of expected affine transformations. While the second conditions impose that the location of the \( \sigma(j) \)th part is consistent with the prediction made from parts \( \sigma(1), \ldots, \sigma(j-1) \). In practice a configuration is deemed a spatially plausible configuration if

\[
 p(\hat{A}) p(x'_{\sigma(1)}|x_{\sigma(1)}), x'_{\sigma(2)}|\hat{A}) \geq \epsilon_A \quad \text{and} \quad
 p(x_{\sigma(2)}|x_{\sigma(1)}), \ldots, x'_{\sigma(j-1)}|\hat{A}) \geq \epsilon_j, \ j = 4, \ldots, P
\]

and where \( \epsilon_A > 0 \) and each \( \epsilon_j > 0 \) is small.

### 2.2. Learning

We have introduced the model to describe the relative spatial relationship between the parts. We now focus on the issue of estimating and learning the parameters of this model. There are two distinct problems:

- Estimation of the parameters \( \alpha, \beta_k \)'s and \( \Sigma \) in eqn (2).
- Estimation of permutations \( \sigma \), the order in which the decomposition in eqn (6) should be performed.

#### 2.2.1 Estimation of the conditional distribution parameters.

We have \( T \) training images with each one containing an instance of the object class. The location of the \( P \) parts are marked in each image and they define \( T \) training configurations \( \lambda^t \) = \( \{x_{t,1}, \ldots, x_{t,P}\} \). It is assumed an affine transformation and an as of yet unspecified shape variation account for the differences between individual configurations. Equation (4) dictates that the parameters of the conditional distribution have to be learned from data varying only by shape variation as opposed to also affine transformation. With this in mind one of the training configurations is chosen as a reference configuration. For all the other training configurations, an affine transformation is applied to warp the configuration as closely as possible (in a least-squares sense) to the reference configuration. Figure 1(c) displays the affine normalized training configurations obtained from the CalTech101 cup database. It is with this data the linear dependencies between the parts are learned. With a temporary (this subsection) abuse of notation the affine normalized configurations will be denoted by \( \lambda^t \). Ideally we want \( \alpha \) in equation (2) to be close to zero to ensure the mean of the Normal distribution is invariant to affine transformations. Therefore, the optimal \( \beta \) for a given \( \sigma \) and \( j \) is defined as:

\[
 \beta_{\sigma,j} = \arg \min_{\beta} \sum_{t=1}^{T} \left\| \sum_{k=1}^{j-1} \beta_k x^t_{\sigma(k)} - x^t_{\sigma(j)} \right\| \tag{9}
\]

subject to \( \sum_{k=1}^{j-1} \beta_{\sigma,j}^k = 1 \). This is a convex optimization problem and can be solved by linear programming.

The estimation of \( \alpha_{\sigma,j} \) and \( \Sigma_{\sigma,j} \) involves a leave-one-out strategy. One of the training configurations \( \lambda^t \) is selected and from the remaining \( T-1 \) training configurations an estimate of \( \beta_{\sigma,j} \) is computed. This estimate is then used to predict the location \( \hat{x}^t_{\sigma(j)} \) in the omitted image and creates a sample of the error term in the linear prediction:

\[
 \hat{x}^t_{\sigma(j)} = \sum_{k=1}^{j-1} \beta_{\sigma,j}^k x^t_{\sigma(j)}, \quad e^t_{\sigma(j)} = \hat{x}^t_{\sigma(j)} - x^t_{\sigma(j)} \tag{10}
\]

This process is repeated for each configuration in the training data and generates a set of sample error terms \( \{e^t_{\sigma(j)}\}_{t=1}^{T} \). The sample mean and covariance of this set are calculated and set equal to \( \alpha_{\sigma,j} \) and \( \Sigma_{\sigma,j} \) respectively. Figure 3(a) shows conditional distributions learned in this fashion.

#### 2.2.2 Identifying good orderings of the decomposition.

It is clear that equation (1) is not particularly valid for all combinations of \( \sigma \) and \( j \). For example, consider the cup in figure 2, it is hard to accurately locate part 1 from parts 3, 4, and 5 as these parts contain no real information about the height of the cup while the location of part 3 is clearly defined by the positions of parts 1, 2, and 4. Therefore, some process is required to decide upon valid \( \sigma \)'s. The following strategy is applied. Each triplet \( (i_1, i_2, i_3) \) (order not important) is considered as a starting point for the permutation describing the order of the decomposition in equation (6) with \( \sigma(1) = i_1, \sigma(2) = i_2, \sigma(3) = i_3 \). For each training image \( t \) the linear transformation \( A_t \) is estimated from the correspondences to the reference configuration \( x'_{\sigma(j)} \rightarrow x_{\sigma(j)}(j = 1, 2, 3) \). It is then investigated if the location of any of the other remaining
parts, $\sigma(4) = k \in \{1,\ldots,P\}\setminus\{i_1, i_2, i_3\}$ can be adequately predicted using the learned conditional distribution $p(x_{\sigma(4)}|x_{\sigma(1)}, x_{\sigma(2)}, x_{\sigma(3)}, A_k)$ across a range of training images. If this is the case then $\sigma(4)$ is set to the part most reliably predicted. Where reliability is a trade-off between the accuracy of the prediction with respect to the covariance the size of the eigenvalues of the covariance. The process is then repeated to find $\sigma(5)$ with $\hat{A}_v$ being updated using 4 correspondences instead of 3 and so on until either $\sigma(P)$ is defined or there is no part that can be adequately predicted by previous parts. This process is somewhat computationally intensive and requires potentially learning up to $\sum_{l=3}^{P-2} \binom{P}{l}$ different conditional distributions. However, this is still computationally feasible for up to around 15 parts. Though, of course, the average computation for objects could be reduced by searching for one feasible $\sigma$ as opposed to all (or the best) $\sigma$.

2.3. Finding plausible configurations

At this stage we have learned a decomposition of the joint probability distribution into simple conditional probability distributions summarizing the relative spatial arrangement of parts by exploiting linear dependencies. This decomposition has been designed so that spatially plausible configurations can be efficiently identified.

Each part has a detector associated with it and when one of these is applied to a novel image it generates, in general, multiple detections, see figure 4(a). The locations of the detections for each part $p$ are denoted by $\{d_p^k\}_{k=1}^{m_p}$. The goal is to find combinations of the detections that could plausibly have been generated by an instance of the target object class, see figure 1. To mathematically formalize this last statement let $g \in \{1,\ldots,m_1\} \times \{1,\ldots,m_2\} \times \cdots \times \{1,\ldots,m_P\}$ be a vector that indicates which detection is being considered for each part. Each $g$ defines a configuration $X_g = \{d_1^g, \ldots, d_P^g\}$ which is a spatially plausible configuration if the linear dependencies between these detections is similar to those of the parts in the training data. Based on the theory introduced earlier it is straightforward to describe an algorithm to identify good $g$. Figure 4 illustrates the process while the next paragraph describes it.

The learned permutation $\sigma$ defines an ordering of the parts. The first three parts, $\sigma(1), \sigma(2)$ and $\sigma(3)$ are used to initialize the search. The possible combinations $(d_{\sigma(1)}^{g_1}, d_{\sigma(2)}^{g_2}, d_{\sigma(3)}^{g_3})$, generated by the detections of these parts are exhaustively enumerated. Each initial triplet of detections is treated as follows. Firstly, it is used to produce an estimate of $\hat{A}$. If $\hat{A}$ is deemed consistent with our prior, then the conditional probability distribution of equation (4) is used to predict an elliptical region where the location of part $\sigma(4)$ should be. Let $v$ be the vector of indices of length $s$ such that $d_s^{v_4}$ lies in this predicted region. If $s = 0$ then the search is stopped and the next initial triplet is investigated. Otherwise, $v$ generates $s$ quartets of detections $(d_{\sigma(1)}^{g_1}, d_{\sigma(2)}^{g_2}, d_{\sigma(3)}^{g_3}, d_v^{g_4})$. Each of these is then investigated in turn to see if it can generate a complete set of $P$
detections by the combination of re-estimating $A$ and using the appropriate conditional probability distribution to predict the position of the next part defined by $\sigma$.

There are couple of things to note. Firstly, that effectively an exhaustive enumeration of all the possible $g$'s is performed. The predictive distributions reduce the search by avoiding investigating unreasonable values of $g$. Secondly, it is obvious we have implicitly assumed that all the parts of the object in the image have been detected by the part detectors. And the above algorithm will fail if a part has been missed. This is a serious limitation as, in general, parts will be missed by a detector or may be occluded. However, though not investigated in this paper it is obvious that subsets of parts can be found and the location of the potentially missing part can be relatively accurately located by the conditional distributions.

3. Exploiting connectedness

Configurations of detections in relative spatial arrangements similar to the training data have been identified. We have done so in an affine invariant fashion, thus there will be generally many more found plausible configurations than actually instances of the object in the image, see figure 5. The next step is to score and rank these configurations and discover the ones that correspond to an instance of the object class. One could potentially use $p(x_1, \ldots, x_p)$ but this would favour configurations close to the average configuration. However, this is not the behaviour we want. To proceed further, examination of the image to look for evidence to support or dismiss each found configuration is required. One obvious cue to exploit is the presence or absence of boundary edges. Many of the parts will be purposely defined on or close to a boundary curve of the object and frequently there will be pairs of parts connected by such a curve, see figure 8. At this point one may be tempted to enter into the world of edge linking and continuation to try and ascertain if such a boundary curve exists. Though progress has been made in recent years with respect to boundary edge detection [16], it is still somewhat hard to predict how robustly they will perform in cluttered images and those containing edges with low contrast. With this in mind we propose a model-based approach rather than a data-driven one.

We return to our training images. The pairs of parts linked by a boundary curve are marked to produce a set of tuples $L = \{(s_i, f_i)\}_{i=1}^L$ where $s_i \neq f_i$ and $1 \leq s_i, f_i \leq P$. For each such pair of points a compact model representing the variation in the shape of the boundary curve is learned. This is done as follows. Consider the pair of parts $(s_i, f_i)$. In each image a non-uniform cubic B-spline, with a fixed number of control points and a fixed knot vector, starting at $x_{s_i}^0$ and ending at $x_{f_i}^5$ is defined by hand to follow the boundary curve between these points. This process defines, for each image, $n_c$ control points $\{p_{s_i,1}, \ldots, p_{s_i,n_c}\}$ where $p_{s_i,1} = x_{s_i}^0$ and $p_{s_i,n_c} = x_{f_i}^5$. These control points are subsequently transformed by the unique similarity transform defined by mapping $p_{s_i,1}^j$ to $(-0.5, 0)$ and $p_{s_i,n_c}^j$ to $(0.5, 0)$ to obtain the Bookstein coordinates [6] of $p_{s_i,2}^j, \ldots, p_{s_i,n_c-1}^j$ with respect to $p_{s_i,1}^j$ and $p_{s_i,n_c}^j$. PCA is then performed on the set of Bookstein coordinates and returns a set of eigenvectors $v_{i,l}$ and associated eigenvalues $\lambda_{i,l}$. The $n_c$ eigenvectors with largest eigenvalues, representing 95% of the variance in the training data, are kept and are a compact representation of the shape variation of the boundary curve. The use of Bookstein coordinates ensures shape changes as opposed to rotational and scaling changes dominate the modes of variations. Figure 6 displays the first two modes of variation for two selected boundary curves.

The curve $Q_s(t)$ between two points $x_{s_i}$ and $x_{f_i}$ is written as a weighted sum of cubic spline basis functions and the control points. The vector of the coordinates of the control points, $P_i$, can be written in terms the PCA basis:

$$\begin{bmatrix} p_{s_i,1}^T, \ldots, p_{s_i,n_c}^T \end{bmatrix}^T = R \left( \mu_i + \sum_{l=1}^{n_c} b_{i,l} v_{i,l} \right) + t \quad (11)$$

where $R$ is a rotation matrix and $t$ a translation vector defined by

$$x_{s_i} = R \left( \begin{array}{c} -0.5 \\ 0 \end{array} \right) + t, \quad x_{f_i} = R \left( \begin{array}{c} 0.5 \\ 0 \end{array} \right) + t \quad (12)$$

and $b_{i,l}$ is the projection of the control points warped to their Bookstein coordinates onto $v_{i,l}$.

Unfortunately, with a given configuration, only the endpoints of the curve $x_{s_i}$ and $x_{f_i}$ are set and the intermediary control points are unknown. It is difficult to robustly fit $Q_s(t)$ to an underlying boundary curve camouflaged by clutter. Rather than rely on a iterative optimization scheme, we propose a simple exhaustive search in each of the modes of variation defined by the PCA basis:

$$P_{i,l,k} = R \left( \mu_i + a_k \sqrt{\lambda_{i,l}} v_{i,l} \right) + t, \quad a_k = -2.5 + k\delta$$

Figure 5. A selection of the spatially plausible configurations found from the detections shown in figure 1. The configurations in general differ by some form of affine variation. Without examining the image it is not possible to say which one corresponds to the true configuration. The last column shows the best configuration based on finding the boundary curves between parts.
where \( \delta > 0 \) is the step size of the search and \( k \in \mathbb{N} \) the number of steps. Each \( P_{t,l,k} \) generates a curve \( Q_{t,l,k}(t) \) estimating the underlying curve \( Q_i(t) \). A cost function is then required to find the best fit between the estimated curve and the image \( I \):

\[
\hat{t}_i, \hat{k}_i = \arg \min_{l, k} C(Q_{t,l,k}(t), I) \tag{13}
\]

We will not explain in detail the mathematical form of the cost function \( C(\cdot, \cdot) \). But it can be adequately described in words in conjunction with figure 6(c,d). Points equally spaced along the length of \( Q_{t,l,k}(t) \) are sampled. At each of these points it is possible to calculate the vector normal to the curve due to our parametric model. If \( Q_{t,l,k}(t) \) is a good estimate of the underlying boundary curve then the normal vector will equal the direction of the image gradient of pixels in a region around this point. For pixels in this region whose gradient magnitude is greater than some small threshold their gradient direction is compared to its predicted value and the absolute difference recorded. This information is aggregated. And in turn this information is aggregated along the curve. A penalty term is applied if the curve passes over an area of constant intensity. Finally the score is normalized by the number of points sampled.

Once an estimate for each of the \( |\mathcal{C}| \) boundary curves has been found then we are in a position to define a cost for the found configuration \( \mathcal{X}' = (x_1, \ldots, x_P) \) with respect to the goodness-of-fit of its boundary edges with the image gradient data. We simply use a weighted sum of cost of each of the boundaries, where each \( w_i > 0 \)

\[
C_{\mathcal{X}'}(x_1, \ldots, x_P, I) = \sum_{i=1}^{|\mathcal{C}|} w_i C(Q_{i,\hat{t}_i,\hat{k}_i}(t), I) \tag{14}
\]

One sensible choice for the weights \( w_i \) is that they are proportional to the length of each boundary. Another is that they are all equal and each boundary edge is considered equal regardless of length. Figure 5 displays the result of applying this process and scoring to the set of spatially plausible configurations found for our cup example.

4. Implementation issues

**Reducing computational complexity** The computational feasibility of the proposed approach, to identify spatially plausible configurations of detections, is dependent on \( P \) and the average number of detections per part in an image. When the latter is in the tens \((<100)\) as opposed to the hundreds and \( P < 7 \) then things proceed nicely. However, when there are several hundreds \((>200)\) of detections per part then computational problems begin to occur both in terms of running time and in memory considerations. This is the case for some of the very cluttered images that occur in the ETHZ shape database \([12]\) the database on which most of the experiments were run. Thus the pruning of the exhaustive search based on the conditional distributions is not aggressive enough in these cases. Therefore some slight modifications are required:

- a strict prior, \( p(A) \), can be applied to the class of affine transformations one expects object instances to undergo in the images being examined. Limits can be set about the amount of \( x, y \)-shear, the ratio of scaling in the \( x \) and \( y \) directions and the in plane 2d-rotation.

- instead of enumerating all spatially plausible configurations generated by the detections, a greedy search can be applied. All configurations of \( P' < P \) parts can be enumerated and then can be scored according to the appropriately modified equation \((14)\). The best \( N \) (where \( N \) is determined by computational resources) partial configurations are kept and from these partial configurations of length \( P' + 1 \) can be generated using the conditional distribution and so on.

- an edge map (output of a Canny detector with default parameter settings) can be used to quickly ascertain if there any evidence of the existence of a boundary edge between two detections where one is expected. A very conservative criterion is set but still manages to achieve some degree of pruning in some images.

**Producing final found configurations** The method proposed for finding spatially plausible configurations from a set of detections and then scoring them produces anything from tens to tens of thousands of different configurations. Many of them will overlap significantly. To produce final detections of the object fairly aggressive clustering is performed. A set of final configurations is initialized with the

Figure 6. Modeling and detecting the boundary curves. (a,b) The 1st and 2nd mode of variation for two of the boundary curves. (c) Measuring if a hypothesized curve is consistent with the underlying image data. Normal directions of the curve at points sampled along it are calculated. Each normal direction is compared to the image gradient direction of the pixels in the displayed regions. (d) For each of the blacks curves shown in (a) and (b) the measurement process described in (c) is performed. The red curves are the ones with the smallest discrepancy between the curve normals and the underlying image gradient directions.
lowest cost (best) configuration. Then each of the other configurations is considered in ascending order with respect to their cost. If it overlaps with any of the final set configurations it is discarded otherwise it is added to the final set. A fairly stringent definition of overlap is used.

**Part detection** Testing our part-based model requires part detections and thus we built simple part detectors for each part. In the experiments object parts were detected using a sliding window approach. The scale space pyramid representation of the image was scanned with a window of fixed size. At each location of the window, a SIFT descriptor [15] was computed and passed to a Support Vector Machine, implemented with libsvm [5] with a polynomial kernel to determine if the target part was present in the window. The SVM was trained with positive examples of the target part that were manually selected in a number of training images. Negative examples were acquired by random sampling from a set of background images. From figure 7 it is clear that these detectors produce many false positives except for the car wheel detector as we have chosen part locations whose structure occurs frequently in images. The situation is also somewhat exacerbated by the fact that our part-based model requires all parts of the object to be detected, thus a conservative threshold is applied in the part detection to ensure a low false negative rate. Therefore, explicitly taking care of missing detections in localization of the part-based object model would allow for part detectors with higher false negative rates.

### 5. Results and conclusions

We present the results of part-based models learned for the three object classes shown in figure 8. Each model was trained using the *cup*, *car side* and *bottle* images from the CalTech101[7] and Caltech256 [13] databases. In each training image the location of the parts were marked by hand as were the boundary curves joining parts. From this data each part-based model was learned as well the PCA representation of its boundary curves.

The ETHZ Shape Classes [12] and the TUD Car Sides [1] dataset were used for evaluation. All the ETHZ images were used in all the experiments (except for 12 of them, a mixture of Giraffe and Applelogo images and a Mug image, that produced too many detections and caused memory problems) while the TUD images were only included for the experiment evaluating the *car side* detector. This was done so that comparison with previously published results on the ETHZ database can be made. Figure 7 gives an indication of the number of number of detections per part for the different part detectors in cluttered images. In the experiments a prior was put on the affine transformations to enforce computational feasibility. 2D rotations of the object up to ±40° from the canonical upright position of the reference configuration were permitted while limits were also put on the ratio of scaling in the $x$ and $y$ directions.

In this paper a detection is defined as a true positive if the bounding box it generates intersects with the hand-marked ground truth and the area of this intersection is greater than $p\%$ of both the detection’s bounding box’s area and the ground truth’s bounding box’s area. Results for both $p = 20, 50$ are reported. Figure 9 displays the performance of each of the part-based models in combination with the boundary curve models for the task of object detection. The results are quite satisfactory considering the number of false positives in the detections of the parts and similar to those reported by other authors.

There are several things to note. The first is that the cars are detected with a far lower fppi than the other two classes. This is mainly due to the fact that wheels can be reliably detected without many false positives (both the left and right wheel detections were set equal to the wheel detections). Thus it does not matter that the other car parts produce so many false detections. Secondly, there is a greater degree of variation within the cups than the bottles and this perhaps accounts for the higher level of found false positives. Also the non-detection of bottles and cups can mostly be blamed on the non-detection of individual parts and the algorithm is not expected to work in such cases. A richer final score, measuring more than the discrepancy between the predicted boundary gradient and the actual gradient, would be required to lower the false positive rate in all cases.

**Conclusions** Experiments confirm that a sparse part-based model with relatively simple parts can be used as a key component of a system to locate object categories defined by
shape. In this paper it is used to highlight possible locations of the object. These areas are then subsequently examined by a more flexible shape model based on splines, which avoids data-driven methods such as edge following and linking, that measures if the image data provides evidence of the existence of boundary/connecting curves between connected parts. The part-based models are designed to decouple variations due to affine warps and other forms of shape deformations. This allows sensible priors, independent of training data, to be set to describe the expected range of affine warps to be applied to the objects in the test images. While measuring the degree of remaining shape variation still has to be learned from representative training data. There are many avenues for future development of the ideas in this paper. The most pressing are the modeling of missed detections of parts and the (semi-)automatic learning of the locations and number of parts for an object.

References