EUROGRAPHICS Workshop on Sketch-Based Interfaces and Modeling (2006)
Thomas Stahovich and Mario Costa Sousa (Editors)

Constellation Models for Sketch Recognition

D. Sharon and M. van de Panne

University of British Columbia†

Abstract

Sketch-based modeling shares many of the difficulties of the branch of computer vision that deals with single image interpretation. Most obviously, they must both identify the parts observed in a given 2D drawing or image. We draw on constellation models first proposed in the computer vision literature to develop probabilistic models for object sketches, based on multiple example drawings. These models are then applied to estimate the most-likely labels for a new sketch. A multi-pass branch-and-bound algorithm allows well-formed sketches to be quickly labelled, while still supporting the recognition of more ambiguous sketches. Results are presented for five classes of objects.

1. Introduction

A large-class of sketch-based modeling systems, specifically those involving drawings of objects, diagrams, or maps, must solve a recognition problem. What did the user draw and what does each stroke correspond to? In many cases, this is solved with the help of domain knowledge, such as knowing that a sailboat has a mast and a hull. This recognition problem has a strong parallel with the goals of single-image interpretation in computer vision, an area which has seen significant progress over the past few years.

We apply a constellation or 'pictorial structure' model to the recognition of strokes in sketches of particular classes of objects. The model is designed to capture the structure of a particular class of object and is based on local features, such as the shape or size of a stroke, and pairwise features, such as distances to other known parts. We learn a probabilistic model from example sketches with known stroke labelings. The recognition algorithm determines a maximum-likelihood labeling for an unlabelled sketch by searching through the space of possible label assignments using a multi-pass branch and bound algorithm. Our technique supports flexible object structure by allowing for optional parts. By applying a recognition threshold, extraneous strokes can also be readily identified.

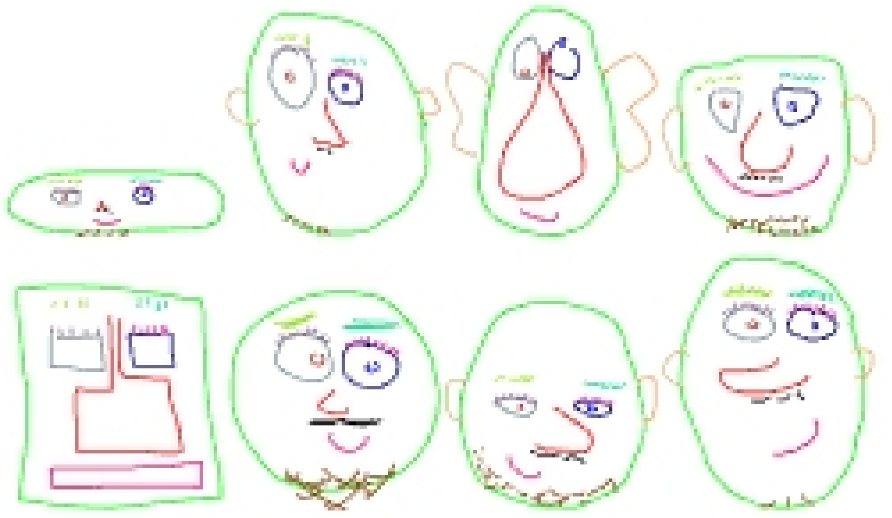
Figure 8 shows an example result for the recognition of parts in face sketches. A subset of the training examples are shown, along with a set of successfully labeled free-form sketches and trace-over sketches. A specific contribution of our method is to cope with objects that exhibit considerable variability in the way they are drawn and that allow a variable number of part instantiations.

The output of our algorithm is a set of labels assigned to the strokes. This can then be utilized by a variety of applications. Labelled strokes can be used to construct parameterized 3D models as in [YSvdP05]. Furthermore, they can help to instance models in a 2D or 3D scene, or serve as a partial interpretation of a larger sketched diagram. Sketches can also be used to retrieve images or 3D models from a database and can, in general, provide an intuitive alternative interface to models with complex internal parameterizations such as faces [fac].

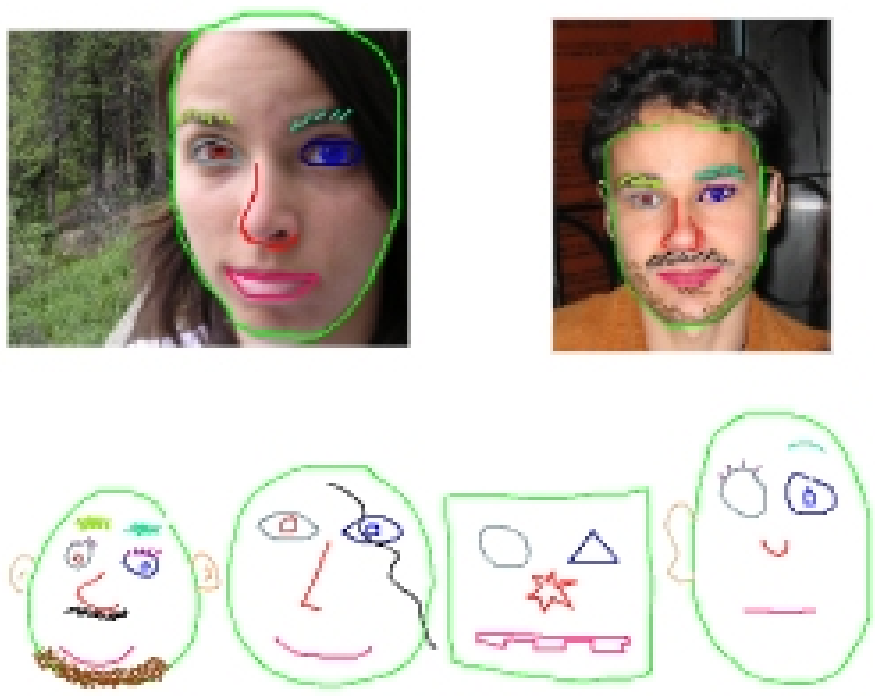
Our system makes two particularly strong assumptions. First, it assumes that similar parts are drawn with similar strokes. For example, a flowerpot that is drawn with four separate strokes instead of one stroke is not easily modelled as part of the same object class. Second, object parts which are deemed mandatory in a sketch must have exactly one instance in the sketch. Optional parts may have multiple instances in a given sketch.

The remainder of the paper is organized as follows. Section 2 gives an overview of related work. Section 3 describes the details of the probabilistic constellation model. Section 4 then describes our algorithms for finding maximum-likelihood interpretations of images using the model. Results are presented and discussed in Section 5, including various

† email: dsharon,van@cs.ubc.ca



(a)



(b)

Figure 1: *(a) Face sketch training examples. The mandatory labels are* head, left-eye, right-eye, mouth, nose*; the optional labels are* left-pupil, right-pupil, left-ear, right-ear, left-eyebrow, right-eyebrow, left-eyelash, right-eyelash, moustache, beard. (b) Face sketches recognized using our system..

modes of use and examples of failure cases. Lastly, Section 6 provides conclusions and future work.

2. Related Work

In this paper we address the problem of understanding completed sketches with a known stroke structure and an unknown stroke ordering. Stroke information is assumed to be collected at the time of drawing creation or it can be extracted using image analysis of a raster drawing using morphology methods (erosion/dilation) and smooth continuation methods.

Recognizing single strokes in isolation is perhaps the simplest version of sketch understanding and can be used to support interfaces that use pen gestures as commands [Rub91]. Recognizing multi-stroke visual structure is significantly more complex, given that the interpretation of strokes is dependant on its local context. Many algorithms use some type of 'parse tree' to search through the space of possible stroke labelings in order to find the most consistent interpretation of a given set of strokes. For applications that involve diagram interpretation, the search is often anchored by first finding well-defined symbols, such as drawn characters or electrical component symbols [KS04]. The search is then further constrained by exploiting the known structure of the given application domain or object classes.

Matching can be treated as a graph isomorphism problem [MF02], where it is applied to the recognition of human stick figures using a known model of connectivity. The work of [YSvdP05] applies a flexible form of hierarchical graph matching. For example, it first looks for the best subgraph representing a cup body before then proceeding to look for optional parts such as cup handles. Curve shape feature vectors are used to quantify the best match and stochastic search is used to explore the space of possible matches. Both of these graph-based models rely heavily on connectivity between parts. They are thus weak at recognizing drawings with disjoint parts, such as a nose or an airplane window.

A probabilistic approach to sketch stroke interpretation is proposed in [AD04]. This uses domain-specific libraries of 'Bayesian network fragments' that describe shapes and domain patterns. Several mechanisms to control the size of the space of hypotheses are presented, and the technique is applied to the domain of electrical circuit diagram recognition. [QSM05] proposes the use of conditional random fields for labeling box-and-line diagrams for particularly difficult ambiguous examples where constraints must propagate in order to find the most-likely interpretation. Perceptually-based shape descriptions are used to help infer the the recognition of image structure in [SMF*02]. Our work looks at recognition problems that do not require connectivity between parts and considers object sketches that can exhibit considerable variability.

Image-based techniques can also be used to help identify sketches or parts of sketches. Shape contexts [BM02] can be used to match sketch images to a fixed set of prototype template images. Image-based classifiers are applied in [SV04] in order to determine likely interpretations for subsets of strokes. An A* search procedure is used to search among the space of possible subset of strokes in order to find a maximum-likelihood interpretation for the image. This is applied to a graphic symbol set of 13 symbols.

Constellation models, also known as pictorial structure models, are composed of a set of local parts, each of which has an appearance model, and a geometry model that defines preferred relative locations or distances of the parts [FE73]. They are well suited to applications such as face recognition, where features such as the nose, eyes, and mouth have particular local features and also have relatively well-defined distances to each other. The model is further developed in [FH05], where it is applied to identify both faces and body configurations from images. The model continues to be extended, with an emphasis on learning pictorial structure models automatically from example images of object classes. More generally, this can be viewed as an example of *statistical relational learning*.

An agent-based approach is presented in [MA03], although this relies on a predefined grammar for the description of the components. The work of [KLP05] is similar to ours in that it uses a constellation-type model and a probabilistic framework. Our work differs in a number of respects, including application to a different domain, using different and larger individual and pairwise feature sets, supporting flexible object classes with optional parts, and a staged search strategy.

Our approach for sketch recognition is uniquely characterized by: (a) support for model definitions derived directly from a set of drawn training examples; (b) a probabilistic framework; (c) support for optional parts; (d) a constellation model with features specifically suited for sketch recognition; and (e) an efficient multi-stage search strategy. We demonstrate our approach on five classes of objects and multiple modes of use (drawing and tracing).

3. The Constellation Model

We represent an object using a constellation model, consisting of features of individual object parts, as well as features of pairs of parts. Individual features capture shape and global positions of parts, whereas pairwise features summarize relative positions of parts. An example constellation model of a face object is shown in Figure 2.

We define a four-element feature vector for individual object parts: $\mathcal{F} = [x\ y\ d\ \beta]$ where (x, y) are the location of the center of the axis-aligned bounding-box (AABB) of a stroke, as measured in image coordinates normalized to $x, y \in [0, 1]$; d is the normalized-coordinate length of the AABB diagonal; and $\beta = cos(\phi)$, with ϕ being the angle of the AABB diagonal with respect to the x-axis.

Similarly, we choose a four-element feature vector for part pairs defined by $\mathcal{G}_{ab} = [\Delta x_{ab}\ \Delta y_{ab}\ D_{ab}\ D_{ba}]$, where $\Delta x = x_a - x_b$ and $\Delta y = y_a - y_b$ define the relative positions of the AABB centers of strokes a and b in normalized coordinates, D_{ab} is the minimum distance between the endpoints of stroke a and any point on stroke b, and D_{ba} is the minimum distance between the endpoints of stroke b and any point on stroke a. In general, $\mathcal{G}_{ab} \neq \mathcal{G}_{ba}$.

Full constellation models do not scale well with the number of parts, n, since they result in $O(n^2)$ pairwise features. We choose to alleviate this by characterizing each label as

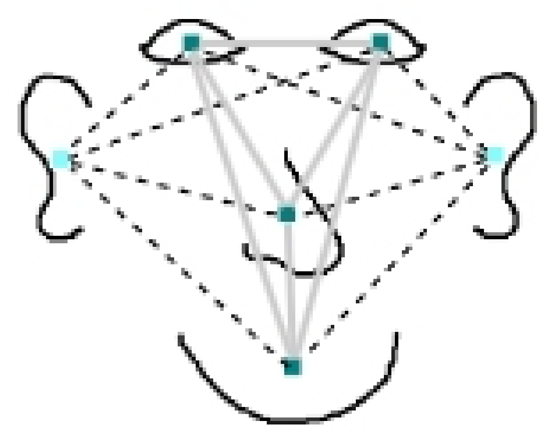


Figure 2: *Example constellation model for a sketched face, showing the pairwise interactions. In this example, the left-eye, right-eye, mouth, and nose are mandatory and thus have complete pairwise interactions. The left-ear and right-ear are optional and thus have pairwise interactions with all mandatory parts but not with each other.*

mandatory or optional. Individual features are computed for both mandatory and optional parts. However, pairwise features are only computed if one or both of the labels in the pair corresponds to a mandatory part. We note that it may be possible to further reduce the number of pairwise features by searching for subsets that yield good recognition performance [CFH05].

The sketch recognition process has two phases, the first which searches the space of possible mandatory label assignments, and the second which searches for optional labels for the remaining unlabelled strokes. In this way the mandatory labels provide contextual location information necessary for assigning appropriate labels to the potentially large number of optional parts. We describe the search algorithm in the following section.

3.1. Learning the Model

An object class model is represented using a probability distribution over features in object constellation models. This function is learned from a set of example labelled sketches. A straightforward choice of object class model is to use multivariate Gaussian distributions. However, in order to support recognition from a small number of training examples, we opt for a diagonal covariance matrix. Thus we independently compute the mean and covariance of each element of the feature vectors $\mathcal{F}$ and $\mathcal{G}$ for the set of labelled sketches that serve as training data. More explicitly, the probabilistic model for the fth element in the feature vector of a label ℓ is given by θ^f_ℓ and consists of the mean value for the feature element, μ^f_ℓ, as well as the standard deviation, σ^f_ℓ. Similarly, a pair feature model, $\theta^f_{\ell j}$, is given by $< \mu^f_{\ell j}, \sigma^f_{\ell j} >$.