Free-hand Sketch Grouping for Video Retrieval

J. P. Collomosse, G. McNeill, and L. Watts
Department of Computer Science, University of Bath, UK
$\{jpc, g.mcneill, l.watts\}$*@cs.bath.ac.uk*

Abstract

We present an algorithm for extracting object descriptions from free-hand sketches of remembered scenes, drawn as video retrieval queries. Our sketches depict scene content, as well as indicators of motion. We report an exploratory study investigating how people sketch to depict recalled events. We incorporate several observations from this study into the design of a novel sketch parsing algorithm. We draw upon a temporal HMM classifier to recognise common pictograms, and graph-cut to identify more general objects.

1 Introduction

Falling hardware costs have prompted an explosion in the creation of digital video assets. The ability to efficiently search repositories to recall video clips is increasingly important, motivating new research into Content Based Video Retrieval (CBVR).

When people recall events, such as those depicted in video, they draw upon their *episodic memory* [13]. Typically, people reconstruct a meaningful, ordered account of an episode from fragments of prior experience. Crucially, people do not literally 'replay' what was actually observed at the time [7]. Conveying this to a retrieval system using keyword (meta-data) queries can be cumbersome. Querying by visual example (e.g. a sketch) is more intuitive, and a number of CBVR systems have been proposed to this end (Sec. 1.1). However these systems do not respect the dynamic and reconstructive nature of episodic memory; rather, they are appearance based, matching on photometric cues such as colour [2] or texture [4] with little concern for object motion.

This paper presents the algorithmic design of a front-end for a novel CBVR system, based on an exploratory study of how people sketch their memories. We accept queries in the form of sketches depicting scene spatial content, alongside cues indicating the *dynamics* (motion) of that content. We refer to this medium as a *storyboard sketch*.

The contribution of this paper is an algorithm for parsing storyboard sketches into descriptions of moving objects. Uniquely, we ground our algorithm in a *qualitative behavioural study* investigating how users sketch to communicate recall of events from episodic memory (Sec. 2). Our findings contrast with prior CBVR, which assumes photometric consistencies between sketches and video that do not typically occur under episodic recall. We also study annotations used to depict motion.

We begin by reporting our empirical study demonstrating how users sketch for event recall (Sec. 2). Using observations derived from that study, we develop an algorithm for parsing storyboard sketches into descriptions of moving objects (Sec. 3). We group sketch strokes into common high-level pictograms using a temporal Hidden Markov Model (HMM) (after Sezgin *et al.* [10]), as well as more general objects using low-level grouping [11]. We report our process for matching this object description to video clips in [8].

1.1 Related Work

Most sketch driven CBR algorithms focus on retrieving static images, but can be extended to CBVR through key-frame extraction. QBIC [2] allows users to sketch queries using blobs of colour. Spatial relationships between regions are used to suggest matches. Other similarity measures include shape and topology [5], texture descriptors [4], and Haar wavelets [3]. Purpose-built CBVR systems use similar measures, but exploit coherence between frames to identify stable features [6].

As with all CBR, we are faced with the task of bridging the "semantic gap" between image processing and the meaningful action in an episode. The semantic gap is particularly interesting in sketch-driven CBVR, since the freedom available to the user *could* lead to a large variation in both query style and content for a given retrieval target. Currently, CBVR systems assume photometric consistency (e.g. colour, texture) across this gap; i.e. between users' sketch representations and the visual properties of a clip. However, this relationship is more complex when users' decisions about the inclusion and juxtaposition of elements in sketches are considered in light of the reconstructive nature of episodic recall.

Besides the content itself, few CBVR systems take into account how users portray motion in sketch queries. Those that do [1, 12] expect queries in the form of coloured polygons, with precise trajectories laid out on the image plane. They assume users can recall the absolute path and speed (e.g. in metres/sec. [1]) of objects.

In this paper, we report sketch-driven CBVR techniques to parse users' sketches of a video episode, based on an improved understanding of how users represent a given video clip when asked to recall it. Sec. 2 describes how our study was conducted and the insights gained. Sec. 3 describes how we translated these insights into a sketch parsing algorithm to narrow the semantic gap.

2 Exploring Episodic Recall Sketches

We devised a study in which participants were shown video clips, and later asked to represent what they could recall about an episode in the video in the form of a sketch. Sketches were then analysed to identify common characteristics among the strategies adopted by users to depict their memories. A total of 14 participants were split evenly into two groups (A/B); each group was shown a different (previously unseen) show-reel of 8 clips over a 2-3 minute period. Participants were inexperienced in sketching by computer. Reel A featured 8 clips of actors engaged in dialogue or stunts. Reel B featured 8 clips primarily of cartoons, sports, and further dialogue. Both reels exhibited clips shot against simple or cluttered backgrounds with a static or panning camera. After a 4 hour delay, participants were asked to recall clips in a randomised order, with the instruction:

"Imagine that you wish to retrieve some video clips from a large video database using a sketch as the query. Most video clips include motion of some type; you should attempt to indicate this motion when you think it is significant."

Participants were provided with a mouse-driven interface (Fig. 1) and were allowed unlimited time both to practice, and then to complete each recall task. Whilst sketching, users were prompted to indicate whether strokes form the foreground, background or a motion cue in their drawings. Sketches were recorded as temporally ordered lists of strokes, including the trajectory (shape and speed) of each stroke, and attributes e.g. color. On completion, discussions were held with participants about the content and their sketching process.

2.1 Summary of Findings

Our discussions and examination of the generated sketches revealed a number of characteristics in users' representations of recalled clips. Despite variations in style, sketch content and level of abstraction was broadly consistent across participants; with similar objects and motions having been depicted for particular clips. For brevity, we have stated the proportion of sketches exhibiting each characteristic. Representative examples are given in Fig. 1. Our main insights are that participants were prepared to compose sketches in terms of foreground and background elements, and that the representational strategies for each of these were distinct in terms of both composition and abstraction characteristics. There were also commonalities in the approaches used to sketch temporal aspects of episodes.

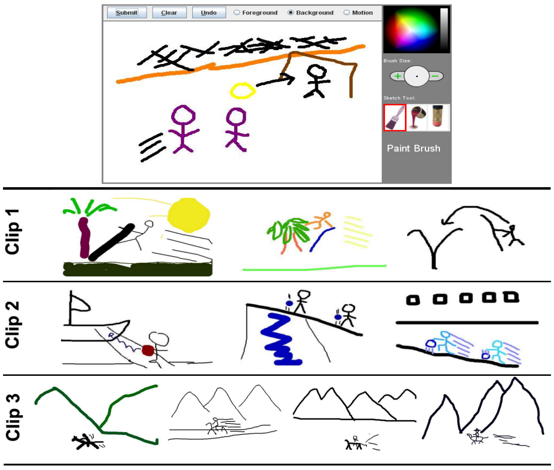

Figure 1. The sketching interface, and representative user sketches of 3 clips.

A. Representation of Recalled Foreground Entities Users draw objects as 'foreground' if they participate in actions of significance, or exhibit large-scale motion relative to the "camera". Inclusion of objects is highly selective; few feature in each sketch. Elements not directly involved in the episode tend to be omitted, regardless of visual salience. Foreground objects are depicted using outlines (line-art), coarsely approximating shape while conserving coarse spatial relationships. Internal detail is omitted or drawn as additional outlined shapes. Objects tend to be drawn using spatially and temporally close strokes; in only 2% of sketches did a user draw a partial object and then return later to finish it. People were drawn pictographically (as stick men) in 84% of all clips that included actors (the exceptions were portrait close-ups). Coarse shape approximations are used to depict non-pictogram foreground objects. These symbols, coupled with streak-lines and arrow-heads, represent a consistent alphabet of pictograms drawn upon by our sketchers. A few saturated, mutually distinctive colours (max. 5, mean 1.82) were used despite availability of a 24 bit palette. Most foreground objects were not coloured in a way that corresponded to appearance: in 84% of sketches a single colour was used to depict an entire foreground object. Rather, colour was used to discriminate between foreground object groups i.e. one colour to categorise each group.

B. Representation of Recalled Clip Background Background was drawn in only 68% of sketches; when background was uncluttered, semantically significant or when the foreground was very sparse. Background objects were drawn very coarsely, as either simplified geometric shapes (e.g. triangles for hills/mountains) or expanses of colour (often scribbled, despite availability of a fill tool). Only 20% of backgrounds were drawn in colour, but of those 91% well approximated the colour *believed to be present* in the video (e.g. blue sky when often the sky was grey or white). Fast, short marks tended to depict texture e.g. crowds, foliage, or water.

C. Time and Motion in Sketches
Motion is indicated by straight or simple curved trajectories capped with arrows (36% of cues), by streak-lines (59%), and occasionally by edge ghosting (5%) – Fig. 1. Arrows are drawn in front of an object's path; streak-lines trail the object. In 92% of cases, cue strokes are drawn in a consistent direction with respect to motion. Only objects sketched as foreground are decorated with motion cues; motion is sketched relative to the background. Periodic motion was indicated by cues either side of the object. Motion cues were drawn as an uninterrupted sequence of strokes — in 97% of cases this was after the moving object was completed. Although the direction of motion was correctly depicted in 98% of cases, there was no reliable indication of motion magnitude. The scene is usually collapsed into a canonical, front-facing perspective that reconstructs memories of action in the episode. Spatial layout of objects was well approximated; objects tend to be sketched near the start of their trajectories, relative to the background. Sketching was temporally inconsistent; objects co-present in a sketch may not appear simultaneously in a clip.

3 Stroke Grouping Algorithm

Our study was exploratory. It was not our intention to derive an immutable set of rules by which people sketch in CBVR, but to look for consistencies in representational strategy and so gain insights into the sketch-based CBVR semantic gap. Our study indicated that users sketched using a combination of coarse shape approximations and a shared alphabet of pictograms for depicting common objects or motions. In contrast to the assumptions of prior work, our study indicates that users include stereotypical, tokenised motion information in sketches, encoding neither absolute path nor speed realistically in these representations. Sketch parsing proceeds as a two step process: pictogram recognition and grouping of the remaining strokes.

3.1 A HMM for Pictogram Recognition

Following [10], we incorporate the temporal ordering of strokes into pictogram recognition. Since sketched pictograms display greater within-class variation than the diagrams considered in [10], we introduce a modified feature set and model. Also, queries contain many non-pictogram strokes, so rather than attempting to explain the entire sketch, we adopt a greedy approach that terminates when no more pictograms can be identified. The simple structure of arrows and streak-lines can lead to false positives. We mitigate this by prompting users to sketch motion cues with a 'motion ink', which separates the motion and non-motion recognition tasks.

We describe each sketch stroke by a feature vector $\mathbf{x}$ and for each class of pictogram m, we train a model λ_m^H which generates sequences $\mathbf{x}_{1..T}$ with associated (hidden) states $z_{1..T}$. The training sequences have different numbers of strokes and contain examples of one/multiple strokes being used for one/multiple 'parts'. The model used is a modified HMM whereby the emission distribution also depends on the variables at the previous step: $p(\mathbf{x}_t|\mathbf{x}_{t-1}, z_{t-1}, z_t)$. This enables the use of relative features *e.g.* (stroke length)/(previous stroke length), but in contrast to [10], the distribution of these higher order features depends on both z_{t-1} and z_t, whereas 'absolute' features such as stroke curvature only depend on z_t. We also learn a spatial model λ_m^G over global object features for each class. The number of strokes is the only feature used for motion cue recognition, whereas for the stickman class, we use more discriminative features such as bounding-box ratio and extrema – *e.g.* "x-value at maximum y". The likelihood score of a sequence $\mathbf{x}_{1..T}$ for class m is:

$$\frac{\alpha \ln(p(\mathbf{x}_{1..T}|\lambda_m^H))}{T} + (1-\alpha)\ln(p(\mathbf{x}_{1..T}|\lambda_m^G)) \qquad (1)$$

Given a test sketch, we evaluate and rank this score for every contiguous subsequence of strokes for each object class; all sequence lengths present in the relevant training set are considered. If the subsequence with the highest score exceeds a threshold, its strokes are labelled accordingly and overlapping subsequences removed from the list of candidate pictograms. The best remaining subsequence in the list is considered iteratively.

Fig. 2 shows representative results; motion cue direction has been interpreted using simple heuristics. Fig. 2 summarises performance against 30 manually labelled sketches; measured by our generalised Rand index eq. (2) with the added condition that strokes contributing to the score must have the correct label, *e.g.* 'stickman' or 'non-pictogram'. Measuring classification and clustering accuracy penalises both under- and over-segmentation (*e.g.* two stickmen identified as one). Note that a single wrongly classified stroke prevents all associated stroke pairs from contributing to the score and hence near perfect recognitions may score $\sim$85%.

3.2 Object stroke grouping via Graph Cut

Non-pictogram strokes are grouped into objects using graph-cut. We compute an adjacency graph $\mathbf{A}$ over all stroke pairs $\{i \times j\}$ using four normalised heuristics: $\hat{a_{ij}} = \sum_{f=1}^{4} \omega_f x_f(i,j)$. Here, $x_1(i,j)$ is the 'shortest path' between strokes — L_2 distance is measured between the closest points on all stroke pairs, yielding a separate graph of inter-stroke distances. Dijkstra's algorithm is applied to that graph yielding a 'shortest path' between stroke i and j. $x_2(i,j)$ is a binary test for colour equality. $x_3(i,j)$ similarly tests stroke foreground/background; a property derived both from users' labelling, and from shape e.g. scribbles tend strokes toward 'background'. $x_4(i,j)$ is a temporal distance; normalised difference between stroke start times.

Weights ω_f are discovered via *a priori* optimization over ground-truth labelled sketches. For a sketch of n objects comprising s strokes, we define the correctness of a sketch grouping as a generalised Rand index [9]:

$$\sum_{i>j}\left[(o_i = o_j)\,(\hat{o}_i = \hat{o}_j) + \frac{(o_i \neq o_j)\,(\hat{o}_i \neq \hat{o}_j)}{n-1}\right] \qquad (2)$$

where o_i is the ground-truth object labelling of stroke i, and $\hat{o}_i$ the label assigned by the algorithm. We normalise this score over all $p = s(s-1)/2$ stroke pairs.

We average (2) over all training sketches to obtain an score for $\omega_{1..4}$. Nelder-Mead search yields the $\omega = \omega_{1..4}$ maximising (2) under constraint $\omega \cdot \omega^T = 1$.

We cut $\mathbf{A}$ using standard n-way cut to maximise graph energy within each sub-graph, and minimise sum of weights along cut boundaries. This yields a partitioning of $\mathbf{A}$ and a cost, for a given n. We independently compute costs over range $n = [1, s]$ and choose n exhibiting lowest cost as our resultant grouping (Fig. 2).

We construct a simple shape model for each object by fitting an active contour around the object's constituent strokes. Motion pictograms are associated with the closest object (or non-motion pictogram) using a distance transform seeded at the base of arrows or centre of streak-line sets. Objects are then recorded as moving in the direction of the cue (or in many directions for multiple cues, used to indicate periodic motion).

4 Conclusion and Discussion

Most sketch driven CBVR employs matching techniques similar to those of CBIR. However video recall depends on *episodic* memory, with consequences for sketch depiction. We suggest that sketch CBVR will not proceed far without analysis of this query medium. To this end we reported a exploratory user study in to how people actually sketch, and encoded our findings into an innovative sketch parsing algorithm.

Optimizing parameter set ω over typical (Fig. 1) training sketches with well-spaced, distinctively coloured objects led to $\omega = [0.346, 0.279, 0.210, 0.164]$ (used for results in Fig. 2), whereas cluttered (overlapping) objects without much use of colour produced $\omega = [0.160, 0.038, 0.448, 0.426]$; the algorithm biases toward weights that have greater discriminatory power. Fig. 2 (right) shows promising accuracy over 30 test sketches; measured using eq. (2) summed over all objects (non-motion pictograms and grouped strokes).

We make a number of assumptions that could be addressed in future; we assume no mis-labelling of strokes by users; we do not handle edge ghosting, or interspersed pictograms (these are rare, Sec. 2). However, we believe our combined contributions of (i) a study exploring sketch recall in CBVR, (ii) an algorithm for parsing such sketches, will provide a valuable resource for researchers of sketch driven CBVR systems [8].

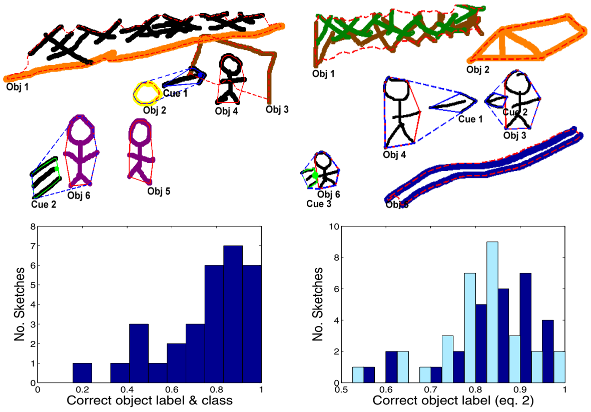

Figure 2. Two groupings (top): solid lines indicate pictograms; dashed lines objects and cue associations. Grouper accuracy, 30 sketches (bot.): HMM pictogram recognition (l.); Cut-based grouper (r., cyan); Combined HMM/cut grouper (r., blue).

References

[1] S. Chang, W. Chen, H. Meng, H. Sundaram, and D. Zhong. VideoQ: an automated content based v ideo search system using visual cues. In *Proc. ACM Intl. Conf. on Multimedia*, pages 313–324, 1997.

[2] J. Hafner, H. S. Sawhney, W. Equitz, M. Flickner, and W. Niblack. Efficient color histogram indexing for quadratic distance. *IEEE PAMI*, 17(7):729–736, 1995.

[3] C. E. Jacobs, A. Finkelstein, and D. H. Salesin. Fast multiresolution image querying. In *Proc. ACM SIGGRAPH*, pages 277–286, Aug. 1995.

[4] S. Kulkarni and B. Verma. Fuzzy logic based texture queries for CBIR. In 5^{th} *Intl. Conf. on Comp. Intelligence and Multimedia Apps.*, page 223, Sept. 2003.

[5] W. H. Leung and T. Chen. Hierachical matching for retrieval of hand-drawn sketches. In *IEEE Conf. on Multimedia and Expo*, volume 2, pages 29–32, July 2003.

[6] R. Lienhart, W. Eiffelsberg, and R. Jain. Visual GREP: A sys tematic analysis of various methods to compare video sequences. In *Storage & Retrieval for Image & Video Databases*, volume 3312, pages 271–282, 1998.

[7] E. Loftus and G. Loftus. On the permanence of stored information in the human brain. *American Psychologist*, 35(5):409–420, May 1980.

[8] G. McNeill and J. Collomosse. Reverse storyboarding for video retrieval. Submitted to ECIR'08.

[9] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Jrnl. ASA*, 66(336):846–850, 1971.

[10] T. Sezgin and R. Davis. Sketch interpretation using multiscale models of temporal patterns. *IEEE Comp. Graphics & Applications.*, 27(1):28–37, 2007.

[11] J. Shi and J. Malik. Normalized cutsand image segmentation. *IEEE Trans. PAMI*, 22(8), August 2000.

[12] C. B. Shim and J. W. Chang. Efficient similar trajectory-based ret rieval for moving objects in video databases. In *Proc. Intl. Conf on Image and Video Retrieval, LNCS.*, volume 2728, pages 158–167, 2003.

[13] E. Tulving. *Elements of Episodic Memory.* Oxford Clarendon, 1983. ISBN: 978-0-19852125-9.