A Computer-Assisted Colorization Approach Based on Efficient Belief Propagation and Graph Matching

Alexandre Noma[1], Luiz Velho[2], and Roberto M. Cesar-Jr[1,⋆]

[1] IME-USP, University of São Paulo, Brazil
alex.noma@gmail.com, cesar@ime.usp.br
[2] IMPA, Instituto de Matemática Pura e Aplicada, Rio de Janeiro, Brazil
lvelho@impa.br

Abstract. Region-based approaches have been proposed to computer-assisted colorization problem, typically using shape similarity and topology relations between regions. Given a colored frame, the objective is to automatically colorize consecutive frames, minimizing the user effort to colorize the remaining regions. We propose a new colorization algorithm based on graph matching, using Belief Propagation to explore the spatial relations between sites through Markov Random Fields. Each frame is represented by a graph with each region being associated to a vertex. A colored frame is chosen as a 'model' and the colors are propagated to uncolored frames by computing a correspondence between regions, exploring the spatial relations between vertices, considering three types of information: adjacency, distance and orientation. Experiments are shown in order to demonstrate the importance of the spatial relations when comparing two graphs with strong deformations and with 'topological' differences.

1 Introduction

Computer-Assisted Cartoon Animation is one of the most challeging areas in the field of Computer Animation. Since the seminal paper of Ed Catmull [4], which describes the 2D animation pipelines and its main problems, intense research has been carried out in this area. Nonetheless, many problems remain unsolved due to several difficulties. On one hand, traditional cartoon animation relies on artistic interpretation of reality. On the other hand, 2D animation relies on the dynamics of an imaginary 3D world, depicted by 2D strokes from the animator. These aspects make most of the tasks in computer-assisted cartoon animation very hard inverse problems, which are ill-posed and closely related to perceptual issues.

Approximations to such problems are required and we may take advantage of structural pattern recognition, which differs from the statistical approach

[⋆] The authors are grateful to FAPESP, CNPq, CAPES and FINEP for financial support, and to Cesar Coelho for the animations used in the experiments.

E. Bayro-Corrochano and J.-O. Eklundh (Eds.): CIARP 2009, LNCS 5856, pp. 345–352, 2009.

because, besides the appearance features, the former explores structural relations between the patterns in order to improve the classification. One of the most important ways to implement structural pattern recognition methods relies on graph representation and matching. Here, we focus on matching two graphs, called model and input graphs. The model graph contains all classes, while the input graph represents the patterns to be classified.

Markov Random Fields (MRFs) have been successfully applied to low level vision problems such as stereo and image restoration. In order to obtain a solution, a cost (or energy) function must be minimized, consisting of the observation and the Markov components [2,7]. The observation component evaluates the appearance (e.g. gray levels of pixels), and the Markov component priviledges particular configurations of labels, simplified to pairwise interactions between sites (e.g. smoothness). Currently, there are two popular approaches to estimate a solution for MRFs: Graph Cuts (GC) [2] and Belief Propagation (BP) [7]. GC are based on an efficient implementation of min-cut / max-flow and BP on a message passing approach in order to propagate appearance and smoothness information through the graph. While the GC based methods are restricted to Markov components representing semi-metrics, the BP based approaches are more general since they do not impose any explicit restriction to the cost function.

A general approach based on graph matching, MRF and BP, with pairwise interactions between sites, is proposed in the present paper for point matching problems. Here, this framework is applied to computer-assisted colorization for cartoon animation [1]. Given an animation sequence, the goal is to track the 2D structural elements (representing regions) throughout the sequence to establish correspondences between the regions from consecutive frames [1] in order to automatically propagate the colors to different frames. This allows fast modifications on the colors of entire animation sequences by simply editing the rendering of one single frame. In this case, each region is represented by its centroid and we want to find a correspondence between two points sets, one from the colored and other from the uncolored frame.

The matching between regions corresponds to a ‘weaker’ form of isomorphism. In practice, the isomorphism is too restrictive and a weaker form of matching is the subgraph isomorphism, which requires that an isomorphism holds between one of the two graphs and a vertex-induced subgraph of the other. An even weaker form of isomorphism is the maximum common subgraph (MCS), which maps a subgraph of the first graph to an isomorphic subgraph of the second one [6].

Closely related to this work is the one due to Caelli and Caetano [3]. They proposed three methods for graph matching based on MRFs, evaluated through artificial experiments to match straight line segments. A key point which has not been explored is the importance that spatial relations can represent, specially when the simplest case of (pairwise) interactions between sites is considered. Here, we extend the efficient BP message computation described in [7], keeping efficiency while exploring three types of structural information simultaneously: adjacency, distance and orientation between patterns. This strategy makes the

Markov component much more discriminative than just 'smoothness'. (Note that in our case, smoothness is also explored by the adjacency between patterns.)

A previous work for computer-assisted colorization was presented in [1], which was based on three factors: region area, point locations and the concept of Degree of Topological Differences (DTD) in order to explore the adjacencies between regions. Here, we propose a simple approach based on area, contour length and point locations used to encode the spatial relations as described in [5] and [8]. The objetive of the proposed general framework based on MRF is to overcome the main difficulty in graph matching problems, expressed by the following question. *How to match two 'topologically' different graphs, with different sizes (different number of vertices and edges), and possibly with 'strong deformations' in terms of appearance / structure between the corresponding patterns?* The proposed method attempts to answer this question by exploring the contextual information, given by the 'labeled' neighbors, through MRFs.

This paper is organized as follows. In Section 2, we formulate the generic graph matching problem as MRF. Section 3 describes the proposed probabilistic optimization approach based on BP. In Section 4, there is a description of our proposed solution to the colorization problem. Section 5 is dedicated to the experimental results. Finally, some conclusions are drawn in Section 6.

2 Graph Matching as MRFs

An Attributed Relational Graph (ARG) $G = (V, E, \mu, \nu)$ is a directed graph where V is the set of vertices of G and $E \subseteq V \times V$ the set of edges. Two vertices $p \in V$, $q \in V$ are adjacent if $(p, q) \in E$. μ assigns an attribute vector to each vertex of V. Similarly, ν assigns an attribute vector to each edge of E. Following the same notation used in [5], we focus on matching two graphs, an input graph G_i, representing the scene (input image) with all patterns to be classified, and a model graph G_m, representing the template with all classes. Given two ARGs, $G_i = (V_i, E_i, \mu_i, \nu_i)$ and $G_m = (V_m, E_m, \mu_m, \nu_m)$, we define a MRF on the input graph G_i. For each input vertex $p \in V_i$, we want to associate a model vertex $\alpha \in V_m$, and the quality of a mapping (or labeling) $f : V_i \rightarrow V_m$ is given by the cost function defined in Equation 1, which must be minimized.

$$E(f) = \sum_{p \in V_i} D_p(f_p) + \lambda_1 \sum_{(p,q) \in E_i} M(f_p, f_q) \ , \tag{1}$$

where λ_1 is a parameter to weight the influence of the Markov component on the result. Each vertex in each ARG has an attribute vector $\mu_i(p)$ in G_i and $\mu_m(\alpha)$ in G_m. The observation component $D_p(f_p)$ compares $\mu_i(p)$ with $\mu_m(f_p)$, assigning a cost which is proportional to the vertices attributes dissimilarity. Each directed edge in each graph has an attribute vector $\nu_i(p, q)$ in G_i and $\nu_m(\alpha, \beta)$ in G_m, where $(p, q) \in E_i$ and $(\alpha, \beta) \in E_m$. The Markov component $M(f_p, f_q)$ compares $\nu_i(p, q)$ and $\nu_m(f_p, f_q)$, assigning a cost which is proportional to the edges attributes dissimilarity.

3 Optimization Based on BP

In order to find a labeling with minimum cost, we use the max-product BP [7], which works by passing messages around the graph according to the connectivity given by the edges. Each message is a vector whose dimension is given by the number of possible labels $|V_m|$. Let m_{pq}^t be the message that vertex p sends to a neighbor q at iteration t. Initially, all entries in m_{pq}^0 are zero and, at each iteration, new messages are computed as defined by Equation 2.

$$m_{pq}^t(f_q) = \min_{f_p} \left(M(f_p, f_q) + D_p(f_p) + \sum_{s \in \mathcal{N}_p \setminus \{q\}} m_{sp}^{t-1}(f_p) \right) \tag{2}$$

where $\mathcal{N}_p \setminus \{q\}$ denotes the neighbors of p except q. After T iterations, a belief vector is computed for each vertex:

$$b_q(f_q) = D_q(f_q) + \sum_{p \in \mathcal{N}_q} m_{pq}^t(f_q) \ . \tag{3}$$

Finally, the label f_q^* which minimizes $b_q(f_q)$ individually at each vertex is selected.

In the following, we describe an efficient computation of each vector message. Equation 2 can be rewritten as [7]:

$$m_{pq}^t(f_q) = \min_{f_p} \left(M(f_p, f_q) + h(f_p) \right) \ , \tag{4}$$

where $h(f_p) = D_p(f_p) + \sum m_{sp}^{t-1}(f_p)$. In order to compute the messages efficiently, based on the Potts model [7], we assume:

$$m_{pq}^t(f_q) = \min \left(H(f_q), \min_{f_p} h(f_p) + d \right) \ . \tag{5}$$

The main difference explored in the present paper relies on $H(f_q)$, which takes into account the edges of the model graph:

$$H(f_q) = \min_{f_p \in \mathcal{N}_{f_q} \cup \{f_q\}} \left(h(f_p) + M(f_p, f_q) \right) \ , \tag{6}$$

where, besides the neighbors in $\mathcal{N}_{f_q}$, it is also necessary to examine the possibility that p and q have the same label.

Thus, to compute each message vector, the amortized time complexity can be upper bounded by the number of edges in the model graph. The standard way to compute a single vector message update is to explicitly minimize Equation 2 over f_p for each choice of f_q, which is quadratic on the number of labels. We propose a modification of the orginal algorithm in [7] to compute each message in linear time on $|E_m|$, which is based on using Equation 6 instead of $H(f_q) = h(f_q)$ for the Potts model described in [7].

4 Computer-Assisted Colorization

The proposed general framework was applied to the computer-assisted colorization problem. Given an animation sequence, the 2D structural elements (representing regions) must be tracked throughout the sequence in order to establish correspondences between the regions from consecutive frames. The goal is to automatically propagate the colors to different frames using this correspondence, which is obtained by a MCS through a cost function, considering two types of information, appearance and structure.

More specifically, given two frames, one colored and the other uncolored, we want to find a correspondence between regions from both frames in order to propagate the colors from the colored to the uncolored frame. Each frame is represented by an ARG. The colored one is the model ARG, while the uncolored one is the input ARG.

Both input and model graphs are obtained similarly. Let W be a set of regions (connected components) defined by the animator strokes in the drawing. Each region in W is represented by its centroid, which is represented by a vertex. Edges are created between adjacent regions, assuming that important contexts are given by adjacent neighbors. Both input and model consist of planar graphs, each one having $|E| = O(|V|)$ edges. Therefore, the algorithm to compute each message vector, described in Section 3, is linear on the number of labels $|V_m|$.

The appearance information is represented by the vertex attributes and the structure by the edge attributes. After T iterations, the BP approach computes the belief vector for each vertex, representing the costs of each label, and assigns a label with minimum cost to obtain a homomorphism. In order to obtain a MCS and to guarantee that the mapping is bijective between subsets of V_i and V_m, we applied the same post-processing as described in [8]: for each model vertex, we kept the cheapest input vertex, and the remaining input vertices were associated to a NULL label, indicating they are not classified, leaving ambiguous cases for the animator to decide which color must be used to the unclassified (uncolored) regions.

Next we describe each term of the energy function in Equation 1.

4.1 Observation Component

For each vertex v, the appearance information $\mu(v)$ consists of two attributes: the area and the contour length of the region corresponding to vertex v. For the observation component, we used

$$D_p(f_p) = \max\left\{\frac{|\mu_A(p) - \mu_A(f_p)|}{\mu_A(p)}, \frac{|\mu_C(p) - \mu_C(f_p)|}{\mu_C(p)}\right\}, \tag{7}$$

where μ_A and μ_C represents the area and the curve length, respectively, $p \in V_i$ and $f_p \in V_m$. In order to map p to f_p, both attributes must match simultaneously, thus leaving ambiguous regions to the user.

4.2 Markov Component

For each directed edge $e \in E$, a single edge attribute $\nu(e)$ is defined as the (normalized) vector corresponding to the directed edge. The Markov component compares edge attributes through the dissimilarity function defined by Equation 8 [5], which compares pairs of vectors in terms of angle and lengths in order to characterize the spatial relations.

$$c_E(\boldsymbol{v_1}, \boldsymbol{v_2}) = \lambda_2 \frac{|\cos\theta - 1|}{2} + (1 - \lambda_2)\big||\boldsymbol{v_1}| - |\boldsymbol{v_2}|\big| , \tag{8}$$

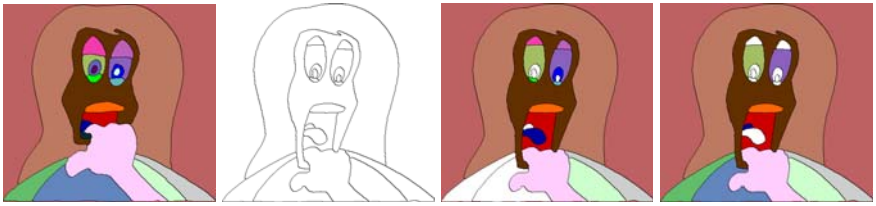
where θ is the angle between the two vectors $\boldsymbol{v_1}$ and $\boldsymbol{v_2}$, $|.|$ denotes the absolute value, $|\boldsymbol{v}|$ denotes the length of $\boldsymbol{v}$ (assuming all lengths $|\boldsymbol{v}|$ are normalized between 0 and 1), and λ_2 is a parameter to weight the importance between the two terms. The Markov component $M(f_p, f_q)$ is defined as the edges dissimilarities described in [5]:

$$M(f_p, f_q) = \begin{cases} c_E\big(\nu_i(p,q), \nu_m(f_p, f_q)\big), \text{ if } (f_p, f_q) \in E_m \\ d, \quad \text{if } (f_p, f_q) \notin E_m \quad \text{and} \quad f_p \neq f_q \end{cases} \tag{9}$$

where the first case compares the respective vectors using Equation 8, and the second penalizes the cost with a positive constant d, encouraging adjacent vertices to have the same label. In this case, $M(f_p, f_q) = M(\alpha, \alpha) = c_E\big(\nu(p,q), \mathbf{0}\big) < d$, proportional to $|\nu(p,q)|$ (because it is assumed $\theta = 0$ in this case), thus penalizing distant vertices. This fact implies that the proposed Markov component is not a semi-metric, since $M(\alpha, \alpha)$ may be different from zero, and the GC [2] based methods are not guaranteed to produce good approximations. Fortunately, this limitation does not apply for the BP algorithm.

5 Experimental Results

The proposed method was tested on four animations: ‘cufa’, ‘calango’, ‘wolf’ and ‘face’. We tested three factors: deformations in appearance due to large



(model) (input) (our result) (previous work)

Fig. 1. ‘Face’ example. From the colored frame (model), we want to colorize the next uncolored input frame. The results from our method and from a previous work. For all colorization experiments, we used $\lambda_2 = 0.5$ (Equation 8), penalty $d = 1.0$ (Equation 9), and all vectors were normalized by the maximum length of all edge attributes (vectors).

Table 1. Quantitative results from the tested animations

animation	frames	$\|V_m\|$	$\|V_i\|$	# wrong colorizations	# missings	# correctly unclassified
cufa	42, 43	35	36	0	4 (11.11%)	2 (100.0%)
calango	24, 25	23	21	1 (4.76%)	3 (14.28%)	1 (100.0%)
wolf	26, 27	23	23	1 (4.35%)	8 (34.78%)	2 (100.0%)
face	–	18	22	0	5 (22.72%)	1 (100.0%)

variations in the region area / contour, deformations in structure due to large motions, and 'topological' differences caused by merging / splitting of regions.

Figures 1 and 2 illustrate the experiments. In all examples, one column illustrates the colored frame for the model and another showing the colorization result. White regions represent the unclassified or uncolored regions.

In Figure 2(a), there is a partial occlusion of the disc, causing one region to disappear and a large change in the 'area'. Also, some regions appear due to splitting of the region near the necklace and medal. Figure 2(b) illustrates a challenging example, with strong deformations on both appearance and structure, merging (e.g. gingiva) and splitting (background) of regions, and new teeths, causing great topological incompatibilities. In Figure 2(c), some regions disappear due to merging of regions (leg and arm).

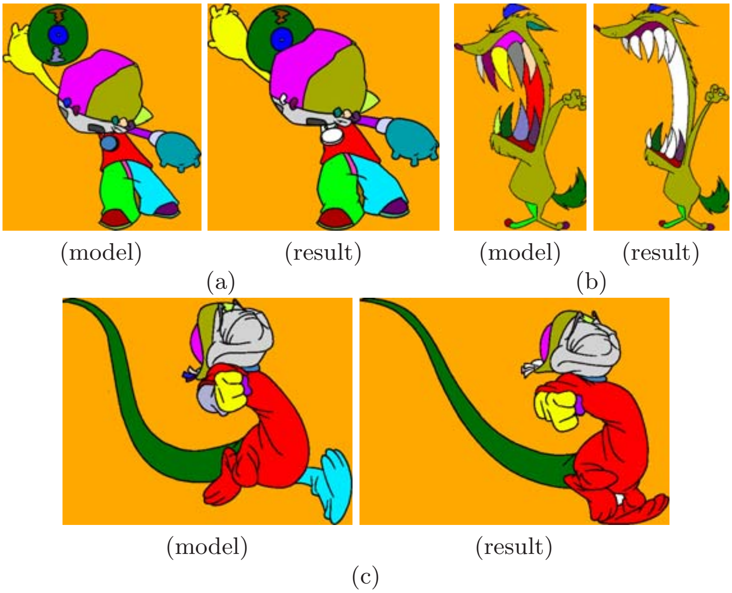

(model) (result)
(a)
(model) (result)
(b)
(model) (result)
(c)

Fig. 2. Example of colorization on the (a) 'cufa', (b) 'wolf' and (c) 'calango' animations. For each example, we present the model and the corresponding result, respectively.

Figure 1 is used to illustrate a comparison against [1]. Quantitative results are shown in Table 1. For instance, in the 'face' example, all the colored regions were correctly matched by our approach (no wrong colorization). Although there were 5 missing regions produced by our approach (and 1 new region correctly unclassified, above the tonge), the method in [1] produced 8 missings (an improvement of 37.5%). Among the 5 missings, 4 were due to changes in the 'adjacency' property, penalized by our approach: two regions of the body and the two pupils. $|V_m|$ and $|V_i|$ denotes de number of model and input vertices, respectively.

6 Conclusions

This paper has proposed a novel general framework for graph matching, using spatial relations through Markov Random Fields (MRFs) and efficient belief propagation (BP) for inference. The edges dissimilarities described in [5] were used as a Markov component, leading to a very useful tool for point matching problems. The key to achieve efficiency was the assumption that important contextual information is concentrated on close neighbors. Both input and model patterns were represented by planar graphs, allowing an efficient algorithm to compute the messages, i.e. linear on the number of labels.

For the computer-assisted colorization problem, we have shown encouraging results, illustrating the benefits of our approach for large deformations in appearance and structure, and for topological incompatibilities on the two graphs being matched, induced by merging and splitting of regions.

Future works include the aplication of the proposed method to other important vision problems, such as image segmentation and shape matching.

References

1. Bezerra, H., Feijo, B., Velho, L.: A Computer-Assisted Colorization Algorithm based on Topological Difference. In: 19th SIBGRAPI, pp. 71–77 (2006)
2. Boykov, Y., Veksler, O., Zabih, R.: Fast Approximate Energy Minimization via Graph Cuts. PAMI 23(11), 1222–1239 (2001)
3. Caelli, T., Caetano, T.: Graphical models for graph matching: approximate models and optimal algorithms. PRL 26(3), 339–346 (2005)
4. Catmull, E.: The problems of computer-assisted animation. SIGGRAPH 12(3), 348–353 (1978)
5. Consularo, L.A., Cesar-Jr, R.M., Bloch, I.: Structural Image Segmentation with Interactive Model Generation. In: ICIP, vol. 6, pp. 45–48 (2007)
6. Conte, D., Foggia, P., Sansone, C., Vento, M.: Thirty Years Of Graph Matching In Pattern Recognition. IJPRAI 18(3), 265–298 (2004)
7. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient Belief Propagation for Early Vision. IJCV 70(1), 41–54 (2006)
8. Noma, A., Pardo, A., Cesar-Jr, R.M.: Structural Matching of 2D Electrophoresis Gels using Graph Models. In: 21st SIBGRAPI, pp. 71–78 (2008)