OBJECT DETECTION IN VIDEO WITH GRAPHICAL MODELS

David Liu *Tsuhan Chen*

Department of Electrical and Computer Engineering, Carnegie Mellon University
Pittsburgh, U.S.A.
{dliu,tsuhan}@cmu.edu

ABSTRACT

In this paper, we propose a general object detection framework which combines the Hidden Markov Model with the Discriminative Random Fields. Recent object detection algorithms have achieved impressive results by using graphical models, such as Markov Random Field. These models, however, have only been applied to two dimensional images. In many scenarios, video is the directly available source rather than images, hence an important information for detecting objects has been omitted — the temporal information. To demonstrate the importance of temporal information, we apply graphical models to the task of text detection in video and compare the result of with and without temporal information. We also show the superiority of the proposed models over simple heuristics such as median filter over time.

1. INTRODUCTION

In images, there exists strong relationship within spatial context. Graphical models such as the Markov Random Fields (MRF)[1] have been used extensively for detection and segmentation. Spatial context can facilitate object detection when the local intrinsic information about the object is insufficient, e.g., when the object appears in a very small scale, or when the object is interfered by background clutter. In video, there exists strong relationship in the temporal context. For object detection, these graphical models, however, have only been applied to images. We show that by applying graphical models over time, we can achieve better results than considering only the spatial context. We also show that this improvement is not easily achieved by ad hoc methods [2] that apply median filter type of rules without statistical analyzing the data.

The graphical model we built our work upon is the Discriminative Random Fields (DRF)[3]. The DRF has been applied to man-made building detection in 2D images [3] and has superior detection ability to the MRF. In Section 2, we briefly review the DRF and extend it from 2D to 3D. In Section 3 and 4, we propose two models which combine the DRF with the HMM. In Section 5 and 6, we use text detection as our testbed, and finally present numerical results and concluding remarks.

Partially supported by the ARDA VACE Program

2. DRF IN TWO AND THREE DIMENSIONS

An input image is partitioned into N_p overlapping patches. A N_f-dimensional feature vector, called *observation*, $\boldsymbol{o}_i \in \mathbb{R}^{N_f}$, $i \in \{1, ..., N_p\}$, is extracted from each patch. The goal is to estimate the corresponding hidden *states* $s_i \in \{-1, +1\}$ (Figure 1 (a)). The DRF [3] has the joint distribution

$$P^{2D}(\boldsymbol{s}|\boldsymbol{o}) = \frac{1}{Z} \exp\left(\sum_i A(s_i, \boldsymbol{o}) + \sum_i \sum_{j \in \mathcal{N}_i} I(s_i, s_j, \boldsymbol{o}) \right) \quad (1)$$

where set $\boldsymbol{s} = \{s_i\}_{i \in \{1,...,N_p\}}$, set $\boldsymbol{o} = \{\boldsymbol{o}_i\}_{i \in \{1,...,N_p\}}$, set $\mathcal{N}_i$ defines the neighbor structure of state s_i, and $Z = Z(\boldsymbol{o})$ is a normalizing constant called the partition function. $A(s_i, \boldsymbol{o})$ is called the association potential, and $I(s_i, s_j, \boldsymbol{o})$ is the interaction potential.

In [3], $A(s_i, \boldsymbol{o})$ is modelled by the logarithm of a logistic regression function, while here we use a Support Vector Machine (SVM) with probabilistic output [4], $P_{\text{SVM}}(s_i|\boldsymbol{o}_i) \in [0, 1]$, so that $A(s_i, \boldsymbol{o}) = log(P_{\text{SVM}}(s_i|\boldsymbol{o}_i))$. We define the interaction potential term as $I(s_i, s_j, \boldsymbol{o}) = \beta s_i s_j$. This term penalizes dissimilar pairs of neighboring states and rewards similar pairs.

We use pseudo-likelihood [1] as an approximation to maximum likelihood. The pseudo-likelihood is the product of the probabilities of states given their neighboring states. The normalization can then be done over single states instead of over the possible configurations of all states. To find the state configuration $\boldsymbol{s}$ given an image, we use the Iterated Conditional Modes (ICM) method [1], which maximizes the conditional posterior probabilities locally in an iterative manner:

$$s_i \leftarrow \arg\max_{s_i} P^{2D}(s_i|s_{\mathcal{N}_i}, \boldsymbol{o}) \quad (2)$$

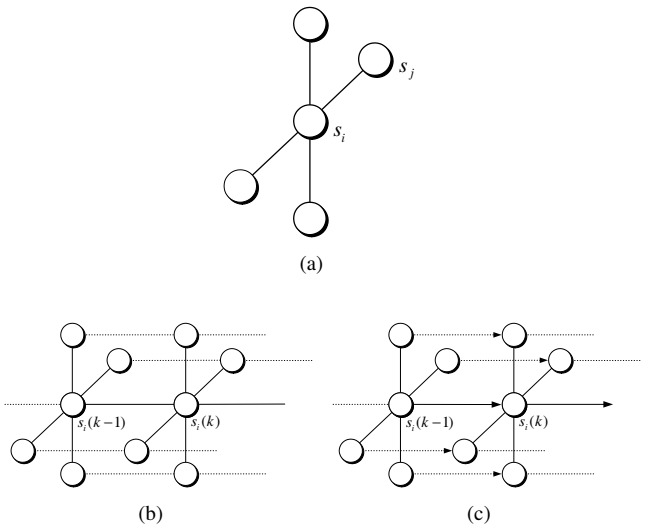

Fig. 1. (a) 2D DRF, with state s_i and one of its neighbors s_j. (b) 3D DRF, with multiple 2D DRFs stacked over time. (c) 2D DRF-HMM type(A), with intra-frame dependencies modelled by undirected DRFs, and inter-frame dependencies modelled by HMMs. States are shared between the two models. In all figures, observations are not shown.

where

$$P^{2D}(s_i|s_{\mathcal{N}_i}, \boldsymbol{o}) = \frac{1}{z_i} \exp\Big(A(s_i, \boldsymbol{o}) + \sum_{j\in\mathcal{N}_i} I(s_i, s_j, \boldsymbol{o})\Big) \tag{3a}$$

$$z_i = \sum_{s_i\in\{-1,+1\}} \exp\Big(A(s_i, \boldsymbol{o}) + \sum_{j\in\mathcal{N}_i} I(s_i, s_j, \boldsymbol{o})\Big). \tag{3b}$$

We initialize the states by using the trained SVM, i.e., $s_i^0 = \arg\max_{s_i} A(s_i, \boldsymbol{o})$.

We extend the 2D DRF to a 3D DRF as follows. We extend the neighboring structure $\mathcal{N}_i$ of each state s_i from 2D to 3D, as in Figure 1 (b). We call neighbors in the same frame as intra-frame neighbors, $\mathcal{N}_i^{\text{intra}}$, and neighbors across neighboring frames as inter-frame neighbors, $\mathcal{N}_i^{\text{inter}}$. Anisotropy for inter- and intra-frame is a natural requirement since dependencies along the temporal direction should be different from the spatial domain, hence define $I^{\text{intra}}(s_i, s_j, \boldsymbol{o}) = \beta^{\text{intra}} s_i s_j$ and $I^{\text{inter}}(s_i, s_j, \boldsymbol{o}) = \beta^{\text{inter}} s_i s_j$. The 3D DRF is in essence collecting more context than the 2D DRF. It therefore has a larger chance to correctly estimate the hidden states.

3. 2D DRF-HMM (A)

In this and the following section, temporal context is modelled by a 1D HMM with discrete output. The HMM connects states across neighboring frames. The number of full connections across two neighboring frames is $O(N_p^2)$, where N_p is the number of states in one frame. Since we allow overlapping image patches, for an 704×480 image, N_p can be as large as 10^5. N_p^2 would then be of order 10^{10}, prohibitively large for HMM decoding. Without loss of generality, we make the simplification that each state is connected only to the state at the neighboring frame with the same position, yielding N_p separate HMMs. The states are shared between the 2D DRFs and 1D HMMs, as shown in Figure 1 (c).

Different from the 3D DRF where the inference is *explicitly* three dimensional, here we decouple the inference between the spatial 2D DRF and the temporal 1D HMM. In other words, the inference is iterated between these two models, which *implicitly* achieves three dimensional inference.

The HMMs use the conditional probability of the 2D DRFs as its *observation*. More precisely, for frame k patch i, the 2D DRF computes $P_k^{2D}(s_i|\boldsymbol{o})$; based on that, the i^{th} HMM has its k^{th} observation as

$$o_i^{\text{HMM}}(k) = \begin{cases} 1, & \text{if} \quad P_k^{2D}(s_i(k)|\boldsymbol{o}(k)) \geq 0.5 \\ 0, & \text{if} \quad P_k^{2D}(s_i(k)|\boldsymbol{o}(k)) < 0.5 \end{cases} \tag{4}$$

Note that observation $o_i^{\text{HMM}}(k)$ is a scalar, and is different from the 2D DRF observation $\boldsymbol{o_i}(k)$, which is a feature vector extracted directly from the image patch.

Each of the N_p HMMs aim at finding the most probable sequence of states given the observation sequence $\{o_i^{\text{HMM}}(1^{st}$ frame), ..., o_i^{HMM}(last frame)$\}, \forall i \in \{1, ..., N_p\}$. Here in the maximization step we use the Viterbi algorithm, which is a maximum-likelihood procedure. After Viterbi decoding, the states are discretized and randomness disappears. Denote the decoded state $s_i(k)$ at frame k by $\hat{s_i}(k)$, where $\hat{s_i}(k) \in \{-1, +1\}$.

The association potential at state s_i now becomes

$$\hat{A}(k) = log(P_{\text{SVM}}(s_i(k) = \hat{s_i}(k)|\boldsymbol{o}_i)). \tag{5}$$

It gives the log-likelihood of a single state after Viterbi decoding. This association potential is by no means the ground-truth potential, because a single run of 2D DRF inference followed by 1D HMM inference still leaves much to be desired.

The 2D DRF-HMM (A) algorithm runs iteratively by executing (6a)-(6d):

$$s_i(k) \leftarrow \arg\max_{s_i} \frac{1}{z_i} \exp\big(A(k) + \sum_j I(k)\big) \tag{6a}$$

$$o_i^{\text{HMM}}(k) = \begin{cases} 1, & \text{if} \quad P_k^{2D}(s_i(k)|\boldsymbol{o}(k)) \geq 0.5 \\ 0, & \text{if} \quad P_k^{2D}(s_i(k)|\boldsymbol{o}(k)) < 0.5 \end{cases} \tag{6b}$$

$$s_i(k) \leftarrow \text{Viterbi}\big(\text{whole sequence}\{o_i^{\text{HMM}}(.)\}\big) \tag{6c}$$

$$A(k) \leftarrow \begin{cases} log(\min(1, \exp(A(k)) + \delta)), \text{ if } s_i(k) = +1 \\ log(\max(0, \exp(A(k)) - \delta)), \text{if } s_i(k) = -1 \end{cases} \tag{6d}$$

where Equation (6a) is a shorthand of Equation (2)(3). Equation (6d) is then followed by Equation (6a). The SVM is used only in the first iteration. Afterwards, the association potential is updated according to the result of HMM decoding as in Equation (6d). Iteration is terminated when states converge.

Experiment shows that gradually updating $A(k)$ as in (6d) yields better result than hard assigning $\exp(A)$ to 1 or 0. The δ in Equation (6d) is a small constant. A possible refinement of adjusting δ is to decrease its value over iterations.

4. 2D DRF-HMM (B)

In the previous section, the 1D HMM shares the 2-state $s_i \in \{-1, +1\}$ with the 2D DRF. Although a 2-state model is sufficient for representing the fact of "object present" and "object absent", we can enrich our model and knowledge representation over the data by considering the following definition of a *track*: A track, $s_i(m:n)$, is a contiguous set of states from frame m to frame n, estimated from a contiguous set of observations $o_i^{\text{HMM}}(m:n)$. It satisfies the condition that $s_i(m:n)$ all say that the object is present, and $s_i(m-1)$ and $s_i(n+1)$ say that the object is absent. In other words, a track represents the life of an object from appearance to disappearance.

To make this explicit, we define a 4-state HMM, $s_i^{\text{HMM}}(k) \in \{0, 1, 2, 3\}$:

1. $s_i^{\text{HMM}}(k) = 0$: $o_i(k)$ comes from non-object.

2. $s_i^{\text{HMM}}(k) = 1$: $o_i(k)$ comes from the start of a track.

3. $s_i^{\text{HMM}}(k) = 2$: $o_i(k)$ comes from an established track.

4. $s_i^{\text{HMM}}(k) = 3$: $o_i(k)$ comes from the end of a track.

The transition probability matrix for the 4-state model clearly has several constraints: transitions from state 0 to state 2, state 0 to state 3, state 1 to state 1, state 2 to state 0, and many other transitions are not allowed.

The 2D DRF-HMM (B) algorithm runs Equation (6a) to Equation (6d) iteratively, except that since the states are no longer shared between the DRF and HMM, we modify Equation (6c) to:

$$s_i^{\text{HMM}}(k) \leftarrow \text{Viterbi}\big(\text{whole sequence}\{o_i^{\text{HMM}}(.)\}\big) \quad (7)$$

and Equation (6d) to

$$A(k) \leftarrow \begin{cases} log(\min(1, \exp(A(k)) + \delta)), \text{ if } s_i \in \{1, 2, 3\} \\ log(\max(0, \exp(A(k)) - \delta)), \text{if } s_i = 0 \end{cases} \quad (8)$$

We further extend the 4-state model by allowing the observations to cover a broader temporal context. While the observations in Section 3 have no overlap, here we allow each observation to overlap with its neighboring (previous and next) observations. Since the observations are discrete, we are essentially increasing the observations from 1-bit to 3-bits, yielding a 4-state 8-observation model. One can naturally think of increasing the number of states or increasing the overlap between the observations even further; however, the amount of training data will finally dictate how many free parameters we can have. It is worth noting that the 4-state model gives a natural semantic explanation of the underlying process, which would be beneficial if we want to explicitly model the duration of tracks.

5. EXPERIMENTAL SETUP

The dataset we use in our experiments is annotated MPEG-2 video obtained from the Linguistic Data Consortium (LDC)[5]. 500 I-frames are used for training, and 4500 for testing. We decode all video frames into color images in RGB format. The resolution of these 5000 images is 704×480.

We use the detection result of the SVM as baseline. The SVM is used in the association potential in 2D DRF, hence the baseline algorithm is essentially the same as the 2D DRF without the interaction term. By using this form of baseline, we can easily see the improvement or degradation of the graphical models which are built upon the SVM. We follow the same procedure as in Section 4.1 in [6] to obtain candidate text bounding boxes. These boxes are scaled and partitioned into overlapping 16×16 pixel patches, with a slide step of 4 pixels. $N_f = 13$ features are extracted from each image patch.

6. NUMERICAL RESULTS

We use the performance measure in [7]. Suppose the detection algorithm returns N_D objects, $D_i, i \in \{1, ..., N_D\}$, and the image actually contains N_G ground truth objects, $G_j, j \in \{1, ..., N_G\}$. A one-to-one mapping $\phi(.)$ between D_i and G_j is found so that the following score over a *single* frame is maximized:

$$Score = \frac{AreaOverlapRatio}{\frac{N_G + N_D}{2}} \quad (9)$$

where

$$AreaOverlapRatio = \sum_i \frac{G_i \cap D_{\phi(i)}}{G_i \cup D_{\phi(i)}} \quad (10)$$

The final performance measure over the *entire* image sequence is taken as the ratio of the sum of the individual scores over the number of frames.

The inference time for the 2D DRF is around 1 second per frame, and around 3 seconds for the 3D DRF and 2D DRF-HMM (A) and (B). Time does not include pre-processing and computing the image features. Implementation is using MATLAB on a Xeon 3 GHz machine.

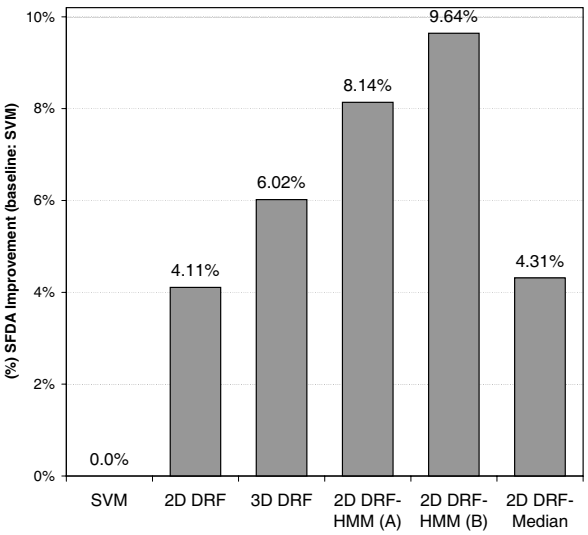

Fig. 2. Comparison of the graphical models with the baseline method. Vertical axis shows improvement in percentage.

We use the score of the SVM as baseline, and compare the graphical models with the baseline. Improvements of scores over the baseline are shown in Figure 2. Figure 3 shows some sample images.

All four graphical models are superior to the baseline SVM. The three graphical models which use temporal information (3D DRF, 2D DRF-HMM (A), and 2D DRF-HMM (B)) show even further improvement over the spatial-only 2D DRF.

The last column, 2D DRF-Median, is the result of iteratively running 2D DRF followed by a median filter over temporal direction on the discrete outputs. More precisely, it also iteratively runs Equations (6a) to (6d), except that the Viterbi algorithm in Equation (6c) is replaced by a median filter. The median filter has a window size of 3. The reason we run this last column is we want to know whether similar performance could be cheaply achieved by 2D DRF followed by simple median filtering. It can be seen that 2D DRF-Median has a slight edge over 2D DRF alone (4.31% vs. 4.11%), while 3D DRF and other temporal models are much better. One reason is that, although the Markovian structure of the DRF in temporal domain enforces conditional independence given the neighbors, the information actually propagates over the entire sequence. This demonstrates the strength of Markov models.

Comparing the 2D DRF-HMM (A) and (B), we conclude that the latter yields better result. Both are superior to the 3D DRF. However, it should be understood that the performance depends on the training and inference methods other than on the model structure alone. Hence, more precisely, by using pseudo-likelihood training and ICM inference for DRF, and using Baum Welch training and Viterbi inference for HMM, we conclude that 2D DRF-HMM (A) and (B) are superior to 3D DRF.

(a) Original image (b) 2D DRF

(c) 3D DRF (d) 2D DRF-HMM (B)

Fig. 3. Sample images.

7. CONCLUSION AND FUTURE WORK

We propose three graphical models to the task of text detection in video. We show that by applying graphical models over temporal context, we can achieve 10% of better results than considering only the spatial context. We also show that this improvement is not easily achieved by ad hoc methods that apply median filter type of rules without statistical analyzing the data. We are currently working on incorporating statistical models into the proposed graphical models to model the duration and number of tracks and false alarms. Higher level representations, such as multi-level HMMs used in the speech recognition community also suggest ways in which our models can be further refined.

8. REFERENCES

[1] S.Z. Li, Markov Random Field Modeling in Image Analysis, Springer-Verlag, 2001.

[2] J. Xi, X.-S. Hua, X.-R. Chen, W. Liu, and H.-J. Zhang, "A video text detection and recognition system," IEEE International Conference on Multimedia and Expo, pp. 873-876, 2001.

[3] S. Kumar and M. Hebert, "Discriminative random fields: A discriminative framework for contextual interaction in classification," IEEE Intl. Conf. Computer Vision (ICCV), Vol. 2, pp. 1150-1157, 2003.

[4] C.-C. Chang and C.-J. Lin, LIBSVM: a library for support vector machines. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`, 2001.

[5] Linguistic Data Consortium. `http://www.ldc.upenn.edu/`.

[6] D. Chen, "Text detection and recognition in images and video sequences," Ph.D. thesis, IDIAP, Switzerland, 2003.

[7] "Performance evaluation protocal for text and face detection and tracking in video analysis and content extraction," VACE program under the Advanced Research and Development Activity (ARDA), 2004.