Reading Report: Video Primal Sketch

Yuanlu Xu
Intelligent Media Computing Lab, SYSU, 2013
merayxu@gmail.com

Abstract

In this report, we discuss a content-based video coding approach, i.e. video primal sketch. Unlike traditional video coding approaches, video primal sketch seeks to capture different mid-level cues and use them to explain the video data. The algorithm first segmented the video data into two different kinds of regions: explicit region and implicit region, and then model each region by a corresponding mid-level video representations. Specifically, explicit region is represented by sparse coding model, where static or moving primitives, such as moving corners, lines, extinguished feature points, are coded by a generalized dictionary. While implicit region is represented by FRAME model, using spatio-temporal filters to match feature statistics, i.e. histograms.

Keywords: mid-level video representation; video primal sketch; sparse coding; FRAME model

1 Introduction

Video coding is an important research orientation due to its huge space cost in transferring and storage. Due to the wide varieties of motion patterns of videos, current video coding techniques represent the video data similar to classical image coding, i.e. treating all the data as uniform signals. However, a key drawback of classical coding methods, such as H.264, JEPG, lies in the little information given by the compressed video codings. That is, we cannot obtain certain video content before decoding a compressed video. Based upon this. content-based video coding aims at discovering content in videos and prioritizing on coding key components. The goal of content-based video coding is not only providing an approach for video compression and coding, but also supporting high-level tasks such as motion tracking and action recognition.

Video primal sketch [12] studies a generic content-based video representation, by integrating two regimes of representations. As illustrated in Fig. 1, an input frame is separated into explicit region and implicit region by binarizing the sketchability map [9, 2] and trackability map [3]. Explicit region including sketchable or trackable parts are modeled by a sparse coding model, and implicit region containing non-sketchable and intrackable parts are synthesized by FRAME model. These two models are integrated in the hybrid representation, i.e. video primal sketch.

The rest of this report is organized as follows. We first present the explicit region representation and sparse coding model in Section 2, and then introduce the implicit region representation and FRAME model in Section 3. The total video representation is summarized in Section 4

2 Explicit Region Representation

In this section, we briefly discuss the explicit region representation, i.e. sparse coding model. The explicit region I_{ex} is decomposed into T_{ex} disjoint video bricks. A video brick is a 11×11 pixels $\times 3$ frames spatio-temporal volume, which record the basic unit of motion. To represent the explicit region, we aim at learning a dictionary W from the

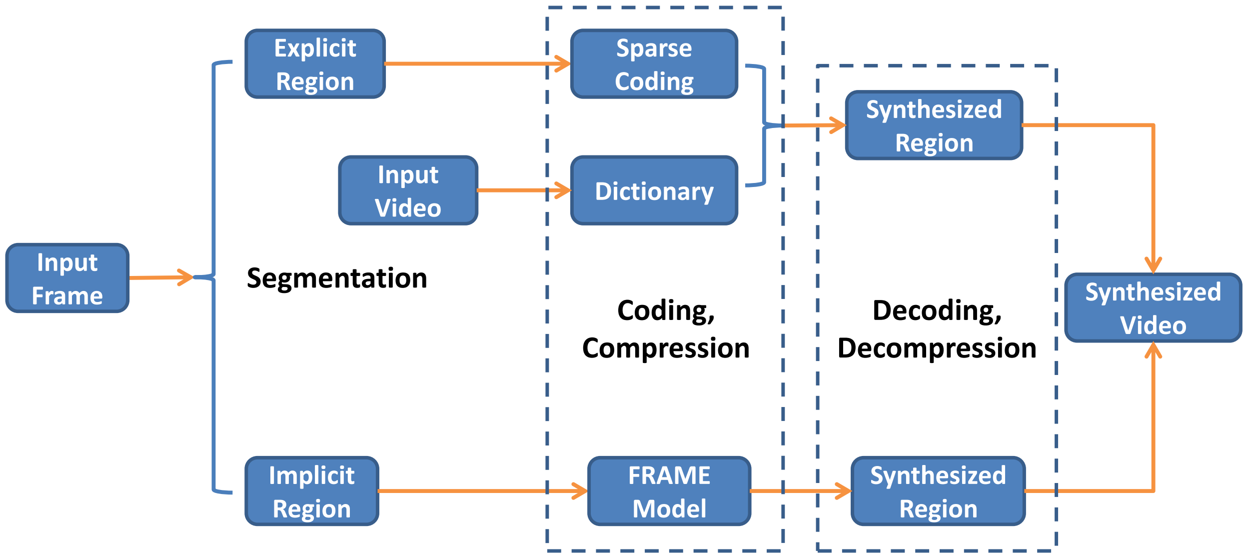

Figure 1: The framework of video primal sketch. Given a input video, the algorithm first segments the video into explicit region and implicit region using sketchability [2] and trackability [3] map. The explicit region is represented by sparse coding model [5] while the implicit region while the implicit region is represented by FRAME model [11]. The synthesized video is obtained by integrating the explicit representations and implicit representations.

training data, which could code the motion variations of all video bricks in the video. In the following discussion of this section, we simplify the notation of the explicit region I_{ex} as I and the number of video bricks T_{ex} as T.

2.1 Linear Generative Model

a. Most models of sparse coding [4, 6, 5] are based on the linear generative model. In this model, given a n-dimensional set of real-numbered input vectors $\nu_i \in \mathbb{R}^n$, the goal of sparse coding is to determine k n-dimensional basis vectors $W = [w_1, \ldots, w_k] \in \mathbb{R}^{n \times k}$ along with a sparse k-dimensional vector of coefficients $\alpha_i = [\alpha_i^1, \ldots, \alpha_i^k] \in \mathbb{R}^k$ for each input vector, so that a linear combination of the basis vectors with proportions given by the coefficients results in a close approximation to the input vector: $\nu_i \approx \sum_{j=1}^{k} w_j \alpha_i^j$.

b. Given a finite training set of input vectors $\{\nu_1, \ldots, \nu_m\}$, the empirical cost function is defined as

$$f_m(D) \triangleq \frac{1}{m} \sum_{i=1}^{m} l(\nu_i, W), \tag{1}$$

where $W \in \mathbb{R}^{n \times k}$ is the dictionary, each column of W representing a basis vector, and l is a loss function such that $l(\nu_i, W)$ measures the coding residual. In general, we have $k \ll m$ and overcomplete dictionary with $k > n$ are allowed. This problem is also known as Lasso [7, 8] in the literature of statistics.

c. $l(\nu_i, W)$ is defined as the optimal value of the l_1-sparse coding problem:

$$l(\nu_i, W) \triangleq \min_{\alpha} \frac{1}{2} \|\nu_i - W\alpha_i\|_2^2 + \theta \|\alpha_i\|_1, \tag{2}$$

where θ is a regularization parameter. Note though l_1 penalty yields a sparse solution for α_i, it is still different from the effective sparsity $\|\alpha_i\|_0$ (i.e. the number of nonzero elements of α_i).

d. A special issue in the lost function is that if W can be arbitrarily large, α would be arbitrarily small. To prevent this, we add constraints to limit each column of W:

$$\mathcal{C} \triangleq \{W \in \mathbb{R}^{n \times k} \, s.t. \, \forall i = 1, \ldots, k, \; w_i^\top w_i \leq 1\}, \tag{3}$$

where $\mathcal{C}$ is called the convex set of dictionaries satisfying this constraints.

e. The problem of minimizing the empirical cost $f_m(W)$ is not convex with respect to W, but it can be rewritten as a joint optimization problem with respect to the dictionary W and the coefficients $\alpha = \{\alpha_1, \ldots, \alpha_T\}$. The problem is not jointly convex but convex with respect to each of the two variables W and α_i when the other is fixed:

$$\min_{W,\alpha} \frac{1}{m} \sum_{i=1}^{m} \left(\frac{1}{2} \|\nu_i - W\alpha_i\|_2^2 + \theta \|\alpha_i\|_1 \right). \tag{4}$$

f. As proposed by Lee et al [4], Equ.(4) can be solved by alternating between the two variables, minimizing over one while keeping the other one fixed.

2.2 Sparse Coding

a. LARS-Lasso algorithm is a regular method to solve the sparse coding coefficients. This algorithm is derived from Lasso algorithm [7], a constrained version of ordinary least squares (OLS), and improved by B.Efron et al. [1] to reduce the computational burden by at least an order of magnitude.

b. LARS algorithm is a forward selection procedure, starting with all coefficients equal to zero. Given an input video brick ν_i, we first find the basis most correlated with response, say w_{i_1}, and take the largest step possible in the direction of this basis until some other basis, say w_{i_2}, has as much correlation with the current residual. LARS then proceeds in a direction equiangular between the two bases until a third basis w_{i_3} enters the "most correlated" set. LARS algorithm then proceeds equiangularly among w_{i_1}, w_{i_2} and w_{i_3}, i.e. along the "least angle direction", etc. In the following discussion of this subsection, we drop the notation of index i of video brick ν_i and coding α_i for notation simplicity.

c. Formally, suppose that $\hat{\alpha}_{\mathcal{A}}$ is the current LARS estimate, i.e. the estimated coding at the $|\mathcal{A}|$-th iteration, the current correlation is thus computed as

$$\hat{c} = W^\top (\nu - W\hat{\alpha}_{\mathcal{W}}), \tag{5}$$

where $\hat{C} = \max_i \{|\hat{c}_i|\}$ denotes the current greatest absolute correlation, $\mathcal{W}$ the set of bases of which correlations equal $\hat{C}$. The next step of LARS algorithm updates $\hat{\alpha}_{\mathcal{W}}$ by

$$\hat{\alpha}_{\mathcal{W}_+} = \hat{\alpha}_{\mathcal{W}} + \hat{\eta}\, u_{\mathcal{W}}, \tag{6}$$

where η is the step length of next step, $u_{\mathcal{W}}$ the direction vector. η and $u_{\mathcal{W}}$ are computed as

$$\begin{aligned} \hat{\eta} &= \min_{w_i \in \mathcal{W}}{}^{+} \left(\frac{\hat{C} - \hat{c}_i}{A_{\mathcal{W}} - W^\top u_{\mathcal{W}}}, \frac{\hat{C} + \hat{c}_i}{A_{\mathcal{W}} + W^\top u_{\mathcal{W}}} \right), \\ u_{\mathcal{W}} &= \mathcal{W} A_{\mathcal{W}} (\mathcal{W}^\top \mathcal{W})^{-1} \mathbf{1}, \end{aligned} \tag{7}$$

where $\min^+$ denotes the minimum only considers positive components and $A_{\mathcal{W}}$ is defined as

$$A_{\mathcal{W}} = (\mathbf{1}^\top (\mathcal{W}^\top \mathcal{W})^{-1} \mathbf{1})^{-\frac{1}{2}}. \tag{8}$$

d. A Lasso solution $\hat{\alpha}$ requires the following restriction

$$\mathrm{sign}(\hat{\alpha}^i) = \mathrm{sign}(\hat{c}_i). \tag{9}$$

However, LARS algorithm does not enforce this restriction. The Lasso modification, i.e. LARS-Lasso algorithm, can yield all LARS solutions to satisfy this restriction. Specifically, in Equ.(6), $\hat{\alpha}_{\mathcal{W}_+}$ will change sign at

$$\eta_i = -\frac{\hat{\alpha}_i}{\mathrm{sign}(\hat{\mathrm{c}}_\mathrm{i}) \cdot (A_{\mathcal{W}} (\mathcal{W}^\top \mathcal{W})^{-1} \mathbf{1})_i}. \tag{10}$$

Let $\tilde{\eta} = \min_{\eta_i > 0} \{\eta_i\}$. The restriction will be violated if $\tilde{\eta}$ is less than $\hat{\eta}$. So the LARS-Lasso algorithm makes the following modifications when the ongoing LARS step reaches $\tilde{\eta} = \hat{\eta}$:

$$\begin{aligned} \hat{\alpha}_{\mathcal{W}_+} &= \hat{\alpha}_{\mathcal{W}} + \tilde{\eta}\, u_{\mathcal{W}}, \\ \mathcal{W}_+ &= \mathcal{W} \setminus w_i. \end{aligned} \tag{11}$$

For detailed explanations and proofs, please check [1].

2.3 Dictionary Learning

a. Classical dictionary learning [6] consists of a sequence of updates of W via projected first-order stochastic gradient descent:

$$W_t = \Pi_{\mathcal{C}} \left(W_{t-1} - \frac{\rho}{t} \nabla_W l(\nu_t, W_{t-1}) \right), \tag{12}$$

where ρ is the learning rate, $\Pi_{\mathcal{C}}$ is the orthogonal projector on $\mathcal{C}$.

b. Online dictionary learning [5] performs stochastic gradient descent on the new sample of the training set and alternates classical sparse coding steps for computing the decomposition of α_t of ν_t over the dictionary W_{t-1} obtained at the the previous iteration, with dictionary update steps where the new dictionary W_t is computed by minimizing over $\mathcal{C}$ the function

$$\hat{f}_t(W) \triangleq \frac{1}{t} \sum_{i=1}^{t} \frac{1}{2} \|\nu_i - W\alpha_i\|_2^2 + \theta \|\alpha_i\|, \tag{13}$$

where coefficients α_i are computed during the previous steps of the algorithm.

Applying the above mechanisms, representing the explicit region is equivalent to learning the dictionary W and computing the codings $\alpha = \{\alpha_1, \ldots, \alpha_T\}$. The process is summarized in Algorithm 1.

Algorithm 1: Representing explicit region with sparse coding

Input: explicit region I, regularization parameter θ, number of video bricks T.
Output: codings $\alpha = \{\alpha_1, \ldots, \alpha_T\}$, learned dictionary $W = [w_1, \ldots, w_k] \in \mathbb{R}^{n \times k}$.

Initialize: dictionary $W_0 \leftarrow 0$, $\Phi_0 \leftarrow 0$, $\Psi_0 \leftarrow 0$;
for $t = 1, \ldots, T$ **do**
  Draw video brick ν_t from I ;
  Compute sparse coding using LARS-Lasso algorithm $\alpha_t \triangleq \arg\min_{\alpha} \frac{1}{2}\|\nu_t - W_{t-1}\alpha\|_2^2 + \theta\|\alpha\|_1$;
  $\Phi_t \leftarrow \Phi_{t-1} + \alpha_t \alpha_t^\top$;
  $\Psi_t \leftarrow \Psi_{t-1} + \nu_t \alpha_t^\top$;
  repeat
    for $i = 1, \ldots, k$ **do**
      Compute dictionary $W_t \triangleq \arg\min_{W} \frac{1}{t} \sum_{i=1}^{t} \frac{1}{2}\|\nu_i - W\alpha_i\|_2^2 + \theta\|\alpha_i\|_1$
        (i) $z_i \leftarrow \frac{1}{\Phi_{ii}}(\psi_i - W_{t-1}\phi_i) + w_{t-1,i}$;
        (ii) Update the i-th column of dictionary $w_{t,i} \leftarrow \frac{1}{\max(\|z_i\|_2, 1)} z_i$;
    end
  until *convergence*;
end

---

3 Implicit Region Representation

In this section, we briefly introduce the implicit region representation. The implicit region I_{im} is first segmented into T_{im} homogenous subregions by taking the explicit parts as boundary conditions. Each subregion $I_{im,i}$ is assumed to be independent and modeled by a FRAME model. In the following discussion of this section, we focus on modeling a single homogenous subregion and simplify the notation of $I_{im,i}$ as I.

3.1 The Minimax Entropy Principle

a. Entropy $\xi(P(I)) = -\int P(I) \log P(I) dI$ stands for the expected coding length. On the other hand, entropy is the negative Kullback-Leibler distance, up to a constant, between $P(I)$ and the uniform distribution, the latter stands for noise images. To minimize the entropy, $P(I)$ should be made as ”orderly” (or far away from the uniform distribution) as possible.

b. To constrain the complexity, we choose an optimal set of features, while it has the minimum entropy. Denoted the feature set S_m, the set of all possible probability distributions $P(I)$ that satisfy the constraints in S_m as Ω_m.

c. The maximum entropy principle suggests that the probability distribution in Ω_m with maximum entropy is the best estimate of $P(I)$.

d. The minimax entropy principle [10] means that $P^*(I)$ should satisfy the constraints and as "orderly" as possible along some dimensions Ω_m, and should also be as random as possible in other unconstrained dimensions.

e. The problem is reformed as follows

$$\begin{aligned} &\text{maximize } -\int P(I)\log P(I)dI, \\ &\text{subject to} \int \phi_i(I)f(I)dI \;=\; \mu_i, \quad i=1,\ldots,m. \end{aligned} \tag{14}$$

According to Lagrange multipliers, the identification of stationary points is

$$\begin{aligned} &\int \phi_i(I)f(I)dI - \mu_i \;=\; 0, \quad i=1,2,\ldots,m \\ &\nabla(-\int P(I)\log P(I)dI) - \sum_{i=1}^{m}\lambda_i\nabla(\int \phi_i(I)f(I)dI - \mu_i) = 0 \end{aligned} \tag{15}$$

The solution $[\lambda_1, \lambda_2, \ldots, \lambda_m]$ is deduced as follows

$$\begin{aligned} &\nabla(-\int P(I)\log P(I)dI) - \sum_{i=1}^{m}\lambda_i\nabla(\int \phi_i(I)f(I)dI - \mu_i) = 0 \\ \Leftrightarrow\; &\frac{\partial(-\int P(I)\log P(I)dI)}{\partial I} - \sum_{i=1}^{m}\lambda_i\frac{\partial(\int \phi_i(I)f(I)dI - \mu_i)}{\partial I} = 0 \\ \Leftrightarrow\; &-P(I)\log P(I) - \sum_{i=1}^{m}\lambda_i\phi_i(I)f(I) = 0 \\ \Leftrightarrow\; &\log P(I) = -\sum_{i=1}^{m}\lambda_i\phi_i(I) \\ \Leftrightarrow\; &P(I) = \frac{1}{Z}e^{-\sum\limits_{i=1}^{m}\lambda_i\phi_i(I)}, \end{aligned} \tag{16}$$

where $Z = \int e^{-\sum\limits_{i=1}^{m}\lambda_i\phi_i(I)}dI$ is the partition function.

Based on simple proof, the Hessian matrix of $\log\, Z$ is the covariance matrix of vector $[\phi_1(I), \phi_2(I), \ldots, \phi_m(I)]$, and is positive definite. Thus, $\log\, Z$ is concave, and the solution for $[\lambda_1, \lambda_2, \ldots, \lambda_m]$ is unique. Considering a closed form solution is not available in general, we seek numerical solutions by solving the following equations iteratively.

$$\frac{d\lambda_i}{dt} \;=\; E_{P(I;\Lambda)}[\phi_i(I)] - \mu_i, \quad i=1,\ldots,m. \tag{17}$$

f. In summary, given the model complexity m, an optimal probability model $p(I)$ or equivalently an optimal probability model $p(I)$ should be derived from the following criterion.

$$P^*(I) = \arg\min_{S_m \in S}\{\arg\max_{P\in\Omega_m}\xi(P(I))\}. \tag{18}$$

3.2 Deriving the FRAME Model

a. To reduce the dimensionality of the distribution $f(I)$. $f(I)$ is transformed into the linear combination of one dimensional marginal distributions. S.C. Zhu et al [11] prove that if the marginal distributions of filter $g_i * I$ for all i are matched, the underlying distribution $f(I)$ can be eventually matched. Considering the complexity of the model, a fixed number of filters are employed to represent $f(I)$.

b. Three assumptions are proposed to further constrain the complexity of FRAME model [11].

1. Discrimination of implicit image region can be captured by the locally supported filters g_i.
2. The implicit image region is homogenous such that $f(I)$ is translation invariant with respect to the pixel location $\vec{p}$.
3. For any probability distribution $P(I)$, if $P(I)$ has the same marginal distribution $f_i(z)$ as $f(I)$, for all $i = 1, 2, \ldots, m$, then $P(I)$ is considered to be perceptually a good enough approximation to $f(I)$.

c. The constraints set $\Omega = \{\, P(I) \,|\, E_P[\delta(I_i(\vec{p}) - z)] = f_i(z),\, \forall z \,\forall i \,\forall \vec{p}\}$ defines that z takes continuous real values, hence there are infinite number of constraints and λ takes the form as a function of z. Assumed that the filter responses I_i are quantified into l discrete values, and the model can be represented as

$$P^*(I) = \frac{1}{Z}\,\exp\left(\sum_{\vec{p}}\sum_{i=1}^{m}\sum_{j=1}^{l}\lambda_i^j \cdot \delta(I_i(\vec{p}) - z_i^j)\right), \tag{19}$$

changing the order of summations, we get

$$P^*(I) = \frac{1}{Z}\,\exp\left(\sum_{i=1}^{m}\sum_{j=1}^{l}\lambda_i^j\, h_i^j\right), \tag{20}$$

where $h_i^j = \sum_{\vec{p}} \delta(I_i(\vec{p}) - z_i^j)$ is the histogram of I_i at the j-th bin. Denoting $h_i = \left[\, h_i^1, h_i^2, \ldots, h_i^l \,\right]$, $\lambda_i = \left[\, \lambda_i^1, \lambda_i^2, \ldots, \lambda_i^l \,\right]$, $P^*(I)$ can be rewritten in a simple parameterized form:

$$P^*(I) = \frac{1}{Z}\, e^{\sum_{i=1}^{m} <\lambda_i,\, h_i>}, \tag{21}$$

which has the following properties:

1. $P(I)$ is specified by $\Lambda = \{\, \lambda_1, \lambda_2, \ldots, \lambda_m \,\}$.
2. Given an image I, its histograms $H = \{\, h_1, h_2, \ldots, h_m \,\}$ are sufficient statistics, i.e. $P(I)$ is a function of $\{\, h_1, h_2, \ldots, h_m \,\}$.

d. Because there is no analytical solution for $\Lambda = \{\, \lambda_1, \lambda_2, \ldots, \lambda_m \,\}$, the numerical solution is obtained by solving the following equation recursively.

$$\frac{d\lambda_i}{dt} = \frac{1}{Z}\,\frac{\partial Z}{\partial \lambda_i} - h_i = E_P(h_i) - h_i, \tag{22}$$

where $E_P(h_i)$ is the expected histogram of the filter responses I_i where implicit image region I follows $P(I; \Lambda)$ with the current Λ.

f. The analytical form of $E_P(h_i)$ is unavailable. To obtain $E_P(h_i)$, a synthesized implicit image region I is sampled by Gibbs sampler given $P(I; \Lambda)$. Hence the histogram h_i of I is used to estimate $E_P(h_i)$.

3.3 The Choice of Filters

a. Assume the estimated probability distribution is $P_i(I)$, Kullback-Leibler distance is applied to measure the difference between $P_i(I)$ and $f(I)$:

$$D(f, P_i) = \int f(I)\,\log\,\frac{f(I)}{P_i(I)} dI = E_f[\log\, f(I)] - E_f[\log P_i(I)]. \tag{23}$$

Based on the definition of entropy, $D(f, P_m)$ can be computed by

$$D(f, P_i) = \xi(P_i(I)) - \xi(f(I)). \tag{24}$$

b. The desired filters are chosen by a stepwise greedy strategy. At the i-th step, Suppose $S_i = \{g_1, g_2, \ldots, g_i\}$ has been selected from the filter bank B. Then at the $(i{+}1)$-th step, the $(i{+}1)$-th filter is chosen from the rest of the filter bank according to the criterion below,

$$g_{i+1} = \arg\max_{g_j \in B \setminus S_i} \frac{1}{2} |\, h_j - \tilde{h}_j \,|. \tag{25}$$

Applying the mechanisms mentioned above, the algorithm of synthesizing an implicit subregion is summarized in Algorithm 2.

Algorithm 2: Synthesizing implicit image region with FRAME model

Input: implicit subregion I, bank of filters B.
Output: probability distribution of implicit image region $P(I)$, synthesized implicit image region $\tilde{I}$.

Initialize: $i \leftarrow 0$, $S_0 = \emptyset$, $P_0(I) \leftarrow$ uniform distribution, $I \leftarrow$ uniform white noise image.
foreach $g_i \in B$ **do**
    Compute I_i by applying g_i to I ;
    Compute histogram h_i of I_i ;
end
repeat
    foreach $g_j \in B \backslash S_i$ **do**
        Compute $\tilde{I}_j$ by applying g_j to $\tilde{I}$;
        Compute histogram $\tilde{h}_j$ of $\tilde{I}_j$;
        $d(j) = \frac{1}{2} |h_j - \tilde{h}_j|$;
    end
    Choose the filter g_{i+1} according to $d(i+1) = \max \{ d(j),\ g_j \in B \backslash S_i \}$;
    $S_{i+1} \leftarrow g_{i+1} \bigcup S_i,\ i \leftarrow i + 1$;
    Initialize $\lambda_j \leftarrow 0,\ j = 1, \ldots, i$;
    repeat
        Calculate $\tilde{h}_j,\ j = 1, 2, \ldots, i$ from $\tilde{I}$;
        Update $\lambda_j,\ j = 1, 2, \ldots, i$ and $P(I; S_i, \Lambda_i)$ is updated ;
        Apply Gibbs sampler to flip $\tilde{I}$ for w sweeps under $P(I; S_i, \Lambda_i)$;
    until $\frac{1}{2} |h_j - \tilde{h}_j| < \epsilon$, *for* $j = 1, 2, \ldots, i$;
until $d(i) < \epsilon$;

4 Hybrid Video Representation

a. In summary, the probabilistic models for explicit region and implicit region are

$$\begin{aligned} I_{ex} &\sim P(I_{ex}; W, \alpha), \\ I_{im} &\sim P(I_{im}; S, \Lambda). \end{aligned} \tag{26}$$

The joint probability model for the video primal sketch representation is defined as

$$P(I|W, S, \alpha, \Lambda) = \frac{1}{Z} \exp \left(-\sum_{i=1}^{T_{ex}} \|\nu_i - W\alpha_i\|^2 - \sum_{i=1}^{T_{im}} \sum_{j=1}^{m} < \lambda_{i,j}, h_{i,j} > \right), \tag{27}$$

where Z is the normalizing constant. Therefore, (W, S) can be viewed as the codebook of the video, while (α, Λ) the compressed codings.

References

[1] B. Efron, T. Hastie, I. Johnstone and R. Tibshirani. Least angle regression. *The Annals of Statistics*, 32(2), 2004.

[2] C. Guo, S.C. Zhu and Y.N. Wu. Primal sketch: Integrating texture and structure. *Computer Vision and Image Understanding*, 106(1), 2007.

[3] H. Gong and S.C. Zhu. Intrackability: Characterizing video statistics and pursuing video representations. *Interational Journal of Computer Vision*, 97(3), 2012.

[4] H. Lee, A. Battle, R. Raina and A.Y. Ng. Efficient sparse coding algorithms. *In Proc. Neutral Information Processing Systems*, 2007.

[5] J. Mairal, F. Bach, J. Ponce and G. Sapiro. Online dictionary learning for sparse coding. *In Proc. International Conference on Machine Learning*, 2009.

[6] M. Aharon and M. Elad. Sparse and redundant modeling of image content using an image-signature dictionary. *SIAM Imaging Sciences*, 1(3), 2008.

[7] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B*, 58(1), 1996.

[8] R. Tibshirani. Regression shrinkage and selection via the lasso - a retrospective. *Journal of the Royal Statistical Society*, 73(3), 2011.

[9] S.C. Zhu, C. Guo, Y.Z. Wang and Z.J. Xu. What are textons? *Interational Journal of Computer Vision*, 62(1-2), 2005.

[10] S.C. Zhu, Y.N. Wu and D. Mumford. Minimax entropy principle and its applications in texture modeling. *Neural Computation*, 9(8), 1997.

[11] S.C. Zhu, Y.N. Wu and D. Mumford. Filters, random field and maximum entropy (frame): Towards a unified theory for texture modeling. *Interational Journal of Computer Vision*, 27(2), 1998.

[12] Z. Han, Z. Xu and S.C. Zhu. Video primal sketch: A generic middle-level representation of video. *In Proc. International Conference on Computer Vision*, 2011.