On segmentation with Markovian models.

Ana Georgina Flesia[1,2], Javier Gimenez[1], Josef Baumgartner[3], ⋆

[1] FAMAF - Universidad Nacional de Córdoba, Medina Allende s/n , Ciudad Universitaria, X5000HUA Córdoba, Argentina.{jgimenez,flesia}@famaf.unc.edu.ar
[2] Conicet at Universidad Tecnólogica Nacional
[3] FCEFyN - Universidad Nacional de Córdoba, Vélez Sarsfield 1611, X5016GCA Córdoba - Argentina. {jbaumgartner}@efn.uncor.edu

Abstract. This paper addresses the image modeling problem under the assumption that images can be represented by 2d order, hidden Markov random fields models. The modeling applications we have in mind comprise pixelwise segmentation of gray-level images coming from the field of Oral Radiographic Differential Diagnosis. Segmentation is achieved by approximations to the solution of the maximum a posteriori equation (MAP) when the emission distribution is assumed the same in all models and the difference lays in the Neighborhood Markovian hypothesis made over the labeling random field. For two algorithms, 2d path-constrained Viterbi training and Potts-ICM we investigate goodness of fit by studying statistical complexity, computational gain, extent of automation, and rate of classification measured with kappa statistic. All code written is provided in a Matlab toolbox available for download from our website, following the Reproducible Research Paradigm.

1 Introduction

Bayesian approaches to image segmentation consists of embedding the problem into a probabilistic framework by modeling each pixel as a random variable, with a given likelihood, embedding the knowledge of the hidden labels on a prior distribution.

The Markov dependence of this prior is defined inside a neighborhood, that is, given all pixels in the neighborhood of a pixel, this pixel is statistically independent of the pixels outside the neighborhood. The result is then obtained by maximizing a Bayesian criterion such as the maximum a posteriori (MAP), see Chen et. al (2012) for a complete review of the field [1]. Estimation over all possible combination of states values is unfeasible in most cases, so only approximated solutions are available in the literature, which are dependent of the original model and the algorithmic choice of restrictions (not always very clear) that are made to deliver an approximation.

For Random Markov Fields, there are available iterative approximated solutions based on the use of the Gibbs distribution as a prior. In Gimenez et

⋆ AGF and JG are partially supported by Secyt-UNC and PICT 2008 00291. JG and JB acknowledge phd grants from Conicet.

al. (2013) [2], a pseudo-likelihood estimator of the smoothness parameter of a second order Potts model was proposed and included on a fast version of Besaj's Iterated Conditioned Modes (ICM) segmentation algorithm. Comparisons with other classical smoothness estimators, like the one introduced on Frery et al (2009) [3], were also provided. Levada et al (2010) [4] also discuss MAP-MRF approximations combining suboptimal solutions based on several different order isotropic Potts priors.

The first analytic solution to a true anisotropic 2-D hidden Markov model (HMM) was introduced by Li et al.(2000) [5]. They studied a strictly-causal, nearest-neighbor, 2-D HMM, and show that exact decoding is not possible. They suggested a decoding approximation called Path Constrained Viterbi training, which was applied to blockwise segmentation of aireal images. Ma et al. (2009) [6] proposed a pseudo noncausal HMM by splitting the noncausal model into multiple causal HMMs, each of which could be solved with PCVA in a distributed computing framework to deliver pixelwise segmentations. Other decoding approximations are discussed in Sargin et al. (2008) [7] and references therein .

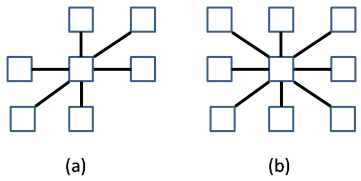

Fig. 1. Neighborhood systems: (a) 2D order Causal Markov Mesh , (b) second order Potts Markov Field, (c) first order Potts model.

In this paper, we explore two specific Markov prior hypothesis for image segmentation, in the form of different neighborhood systems and probability relationships:

1. a diagonal six-pixel neighborhood for the anisotropic MRF
2. an eight-pixel neighborhood for the isotropic MRF

which are depicted in Figure 1. For the first neighborhood system we ensure assumptions of a 2d order Causal Markov Mesh model, that introduced an extra assumption in the neighborhood probabilities related to the notion of "pixel's past". In the second neighborhood, we introduce a Gibbs distribution, in the form of the Potts model with a smoothness parameter β. We keep the same Multivariate Gaussian model for the observations (conditional to the class) in order to ensure the same initial conditions. The MAP approximations implemented on each case were our own version of Jia Li's Path Constrained Viterbi Training (PCVT) [5] for the anisotropic case and Frery's ICM [3] for the isotropic case.

In Section 2 and 3 we discuss in detail the equations of our implementations. In Section 4 we introduce the design of our simulated experiments and the statis-

tics used to evaluate goodness of fit. In the final section we discuss conclusions and prospects.

2 MAP-MRF rules

Many effective computational tools for practical problems in image processing and computer vision have been devised through Markov Random Fields (MRF) modeling. One of these practical problems is to label an image domain pixelwise with given L discrete labels $\mathcal{L} = \{\ell_1, ..., \ell_L\}$, with the help of a priori modeling hypothesis like the Potts model.

Markov random fields provide convenient prior for modeling spatial interactions between pixels. Let $\mathcal{P}$ be a set of pixels in the image I of size $n = z \times w$, $\mathcal{L} = \{\ell_1, \ell_2, \ldots, \ell_L\}$ a set of L labels, and for each site $(i,j) \in \mathcal{P}$ is defined as a set $\partial_{ij} \subset \mathcal{P}$ called the *neighborhood* of (i,j). The neighborhood system is a collection of sets $\partial = \{\partial_{ij} : (i,j) \in \mathcal{P}\}$ which satisfies: (a) $(i,j) \notin \partial_{ij}$,(b) $(i',j') \in \partial_{ij} \Rightarrow (i,j) \in \partial_{i'j'}$, (c) $\mathcal{P} = \bigcup_{(i,j)\in\mathcal{P}} \partial_{ij}$.

The labeling problem is to assign a label from the label set $\mathcal{L}$ to each site in the set of sites $\mathcal{P}$. Thus a labeling is a mapping from $\mathcal{P}$ to $\mathcal{L}$. We will denote a labeling by $s = \{s_{ij}\}$. The set of all possible labeling $\mathcal{L}^n$ is denoted by $\mathcal{S}$.

In this paper we will consider the final segmentation $s = \{s_{ij}\}$ as realizations of a Markov random field. This means that for each possible realization (called configuration) $s \in \mathcal{S}$, it holds that $p(s) > 0$, $p\left(s_{ij}|s_{\mathcal{P}-\{(i,j)\}}\right) = p\left(s_{ij}|s_{\partial_{ij}}\right)$. where $\mathcal{P} - \{(i,j)\}$ denotes set difference, and $s_{\partial_{ij}}$ denotes the labels of the sites in ∂_{ij}.

In general, the labeling field is not directly observable in the experiment. We have to estimate its realized configuration s based on an observation I, which is related to s by means of the likelihood function $p(I|s,\theta)$, where θ represents the set of all model's parameters. The most popular way to estimate an MRF is to maximize a posteriori (MAP) estimation.

MAP estimation consists of maximizing the posterior probability $p(s|I,\theta)$. From the point of view of Bayes estimation, the MAP estimate minimizes the risk under the zero-one cost function. Using Bayes rule, the MAP estimate is

$$s^* = arg\max_{s\in\mathcal{S}} p(s|I,\theta) = arg\max_{s\in\mathcal{S}} p(I|s,\theta)\, p(s|\theta) \tag{1}$$

We assume that the pixel intensities I_{ij} are random vectors from $\mathbb{R}^q$, with Multivariate Gaussian emission probabilities given the state $\ell \in \mathcal{L}$:

$$p(x|\ell) = \frac{1}{(2\pi)^{q/2}|\Sigma_\ell|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu_\ell)^T \Sigma_\ell^{-1}(x-\mu_\ell)\right\} \tag{2}$$

with $x \in \mathbb{R}^q$, mean μ_ℓ and covariance matrix Σ_ℓ.

3 Gibbs prior

MRFs are generalizations of Markov processes that can be specified either by the joint distribution or by the local conditional distributions. However, local

conditional distributions are subject to nontrivial consistency constraints, so the first approach is most commonly used. In this paper, we will consider two types of Markov random Fields as prior constraints for the labeling field, causal Markov Meshes, and Gibbs random fields.

Before defining Gibbs random fields (GRF) we need to define a *clique*. A set of sites is called a clique if each member of the set is a neighbor of all the other members. A Gibbs random field can be specified by the joint Gibbs distribution: $p(s) = Z^{-1} \exp\left(-\sum_{C \in \mathcal{C}} V_C(s)\right)$, where $\mathcal{C}$ is the set of all cliques, Z is the normalizing constant, and $\{V_C : C \in \mathcal{C}\}$ are real functions, called the clique potential functions. In this model, the conditional distribution of state label $s_{ij} \in \mathcal{L}$ corresponding to pixel $(i,j) \in \mathcal{P}$ given the evidence in the image is

$$p(s_{ij}|s_{i'j'} : (i',j') \in \partial_{ij}) = p(s_{ij}|s_{i'j'} : (i',j') \neq (i,j)) \propto \exp\left(-\sum_{C \in \mathcal{C}:(i,j) \in C} V_C(s)\right). \tag{3}$$

This Markovian assumption guarantees the existence of the joint distribution of the process.

3.1 Potts model

In this model, the potential functions V_C of (3) are defined as follows:

$$V_C(s) = \begin{cases} -\beta & \text{if } s_{ij} = s_{i'j'}, C = \{(i,j),(i',j')\} \in \mathcal{C}, \\ 0 & \text{in other case.} \end{cases} \tag{4}$$

where $\mathcal{C}$ is the clique set corresponding to the neighborhood's system ∂. Thus, the distribution on the neighborhood in the Potts model becomes

$$p(s_{ij}|s_{i'j'} : (i',j') \in \partial_{ij}) \propto \exp\{\beta U_{ij}(s_{ij})\}$$

where $U_{ij}(s_{ij}) := \#\{(i',j') \in \partial_{ij} : s_{i'j'} = s_{ij}\}$, and β is the smoothness parameter, sometimes called inverse temperature. Thus, the joint likelihood of the Markov random field is

$$p(s) \propto \exp\{\beta U_s\},$$

where $U_s = \#\{C \in \mathcal{C} : C = \{(i,j),(i',j')\}, s_{ij} = s_{i'j'}\}$.

The observed process, which is supposed to be emitted by the hidden Markov Field, is considered Multivariate Gaussian as in (2) with mean μ_l and covariance matrix Σ_l which depends on the classes. Thus, given the observed pixel intensities I, the a posteriori distribution of the map of classes is

$$p(s|I,\theta) \propto \exp\{\beta U_s + \sum_{ij} p(I_{ij}|s_{ij})\}. \tag{5}$$

This distribution corresponds to a new Potts model in which the external field in a given pixel (i,j) is $p(I_{ij}|s_{ij})$.

The optimum segmentation s^* is defined as a MAP solution (1), with $\theta = (\beta, \mu_l, \Sigma_l)$ which is usually unfeasible. There are many approximated solutions provided in the literature, we will work with the version of Iterated Conditional Modes given by Frery et al (2009)[3].

Iterated Conditional Modes (ICM) is an iterative algorithm that rapidly converges to the local maximum of the function $P(s|I, \theta)$ closest to the initial segmentation provided by the user. Usually, the initial segmentation is provided by Maximum Likelihood. In each iteration ICM modifies the label of each pixel for the label that is most probable, given the neighborhood configuration.

Given observations I, ICM finds a suboptimal solution of (1), with the algorithm 1. The first term of (6) is equivalent to the ones used by the ML classifier.

Algorithm 1: Iterated Conditional Modes (ICM)

1) Maximum Likelihood segmentation of I.
2) Parameter estimation by pseudo-maximum likelihood with Brent's algorithm for the smoothness parameter of the second order isotropic Potts model.
3) Choose a pixel's visit scheme for the image.
4) For each pixel (i, j), change the label given in the previous iteration for the label $\ell \in \mathcal{L}$ that maximizes

$$g(\ell) = \ln p(I_{ij}|\ell, \hat{\mu}_\ell, \hat{\Sigma}_\ell) + \hat{\beta} U_{ij}(\ell) \tag{6}$$

5) Iterate until convergence.

The second term is the contextual component scaled by the parameter β. If $\beta > 0$, ICM smooths out the initial segmentation,if $\beta < 0$, ICM reduces clusters coherence. When $\beta = 0$ the rule is reduced to the maximum likelihood, but when $\beta \to \infty$ the effect is reversed, the data does not have any importance in the final segmentation.

3.2 Causal prior

For pixel (i, j) we define the following causal relationship that represents the "past", $(i^{'}, j^{'}) \prec (i, j)$ if $i^{'} < i$ or $i^{'} = i$ and $j' < j$, and the set $\{s_{i,j-1}, s_{i-1,j}\}$ the neighborhood of (i, j) in the past. We assume a Causal 2D order Markov Mesh model stating that, for state s_{ij}:

$$p(s_{ij}|s_{i'j'} : (i', j') \prec (i, j)) = p(s_{ij}|s_{i,j-1}, s_{i-1,j}). \tag{7}$$

The two pixels $(i, j-1)$ and $(i-1, j)$ can be understood as the "past" of pixel (i, j). We consider $P(s_{ij}|s_{i,j-1}, s_{i-1,j})$ to be independent of the current pixel so we can gather the transition probabilities in a matrix A where

$$a_{\ell_1,\ell_2,\ell_3} = p(s_{ij} = \ell_1 | s_{i,j-1} = \ell_2, s_{i-1,j} = \ell_3) \qquad \forall \ell_1, \ell_2, \ell_3 \in \mathcal{L}.$$

The transition matrix A has one more dimension than the transition matrix of a 1d Markov Process due to two "past states" on the left and above the actual pixel. This yields at a new order of the image. Instead of lining up the pixels as we would have done in the one-dimensional case we are now moving from the top-left pixel to the bottom-right pixel. Thus, the initial probabilities for the 2dMM depend only on the first state $s_{0,0}$ and we can write

$$\pi_\ell = p(s_{0,0} = \ell) \qquad \forall \ell \in \mathcal{L}.$$

The word "hidden" that is usually added to the whole model (Gaussian observed process plus Markov Mesh labeling random field) comes from the fact that this Markov Mesh can not be observed, so it is considered hidden. It can be proved that this causal relationship implies a general Markov Field hypothesis with the diagonal neighborhood stated in the introduction, that is, the probability of a label given the whole labeling specification depends only on the values in the pixels depicted in Figure 1 (a).

If we enumerate each diagonal in the image, $T_0, \ldots, T_{z+w-2}$, as one step in time, starting with the top-left pixel, see Figure 8,

$$T_0 = (s_{0,0});\ T_1 = (s_{1,0}, s_{0,1});\ T_2 = (s_{2,0}, s_{1,1}, s_{0,2});\ \ldots\ ;\ T_{z+w-2} = (s_{z-1,w-1});$$

the Markov Mesh assumption (along with the particular definition of the past) implies that

$$\begin{aligned} p(s) &= p(T_0)p(T_1|T_0)\ldots p(T_{z+w-2}|T_{z+w-3}, \ldots, T_0) \\ &= p(T_0)p(T_1|T_0)\ldots p(T_{z+w-2}|T_{z+w-3}). \end{aligned}$$

This means that each diagonal operates as an "isolating" element between neighboring diagonals, which suggest an extension of the 1d Viterbi algorithm to compute the most probable sequence of states given initial values. This is, to find the optimal combination of states s^* that solves (1) given the whole 2D hidden Markov model, the labeling field and the observed Gaussian intensity process.

Let $T_0 = (s_{0,0});\ \ T_1 = (s_{1,0}, s_{0,1});\ \ \ldots$ be a path through the image where every diagonal marks one step. Each diagonal consists of up to $\min(w, z)$ states: $T_0 \in \mathcal{L},\ T_1 \in \mathcal{L}^2,\ T_2 \in \mathcal{L}^3,\ \ldots,\ T_{z+w-2} \in \mathcal{L}$. This makes a total of $L^{\min(w,z)}$ possible state combinations only considering the main diagonal. Therefore, the exact decoding of our problem is an NP-hard problem. To produce an approximated solution we will work constraining the set of possible state combinations.

3.3 Path-Constrained Viterbi Training

The Viterbi training algorithm is an iterative algorithm that estimates all the parameters of a HMM, and finds the sequence of states that best explains the data, given the estimated parameters. The procedure starts with the setting of initial parameters, which can be done using prior information, educated guess or a non-contextual estimation. Using this initial step the algorithm follows the next steps until convergence

Algorithm 2: Path-Constrained Viterbi Training (PCVT)

- Initialize segmentation $s^{(0)}$: Maximum Likelihood segmentation of I .
- Parameter estimation: Given sequence $s^{(n-1)}$, estimation of $a^{(\mathfrak{n})}_{\ell_1,\ell_2,\ell_3}$, $\mu^{(\mathfrak{n})}_{\ell}$ and $\Sigma^{(\mathfrak{n})}_{\ell}$
- Decoding: Choosing the best N paths and Viterbi decoding using these paths.

First step Viterbi Training: Parameter estimation Let's suppose we have the initial sequence $s^{(0)}$ obtained from maximum likelihood classification, or the sequence $s^{(n-1)}$ obtained from Viterbi in the previous step. Our empirical estimations of the transition probabilities and distributions parameters are

$$a^{(\mathfrak{n})}_{\ell_1,\ell_2,\ell_3} = \frac{\sum_{i=1}^{z-1}\sum_{j=1}^{w-1}\chi\left(s^{(\mathfrak{n}-1)}_{i-1,j}=\ell_1, s^{(\mathfrak{n}-1)}_{i,j-1}=\ell_2, s^{(\mathfrak{n}-1)}_{ij}=\ell_3\right)}{\sum_{i=1}^{z-1}\sum_{j=1}^{w-1}\chi\left(s^{(\mathfrak{n}-1)}_{i-1,j}=\ell_1, s^{(\mathfrak{n}-1)}_{i,j-1}=\ell_2\right)} \tag{8}$$

$$\mu^{(\mathfrak{n})}_{\ell} = \frac{\sum_{i=0}^{z-1}\sum_{j=0}^{w-1}\chi\left(s^{(\mathfrak{n}-1)}_{ij}=\ell\right) I_{ij}}{\sum_{i=0}^{z-1}\sum_{j=0}^{w-1}\chi\left(s^{(\mathfrak{n}-1)}_{ij}=\ell\right)}, \Sigma^{(\mathfrak{n})}_{\ell} = \frac{\sum_{i=0}^{z-1}\sum_{j=0}^{w-1}\chi\left(s^{(\mathfrak{n}-1)}_{ij}=\ell\right)(I_{ij}-\mu_\ell)(I_{ij}-\mu_\ell)^T}{\sum_{i=0}^{z-1}\sum_{j=0}^{w-1}\chi\left(s^{(\mathfrak{n}-1)}_{ij}=\ell\right)} \tag{9}$$

where χ is the indicator function.

Second step Viterbi training: Decoding There are several different approximations in the literature for iterative decoding. Sargin et al (2008)[7] proposed an algorithm that iteratively updates the posterior distribution on rows and columns, i.e. determining horizontal and vertical 1d forward-backward probabilities, combining them to approximate the values of $p(s_{ij}|s_{i,j-1}, s_{i-1,j})$ as product of horizontal and vertical probabilities. A more simplistic approach is to represent the dependency of the neighbors as the horizontal and vertical conditionals, a row and column wise constrained application of belief propagation. Such models deviate us from the original Makovian assumptions, so in this paper we will follow the so called Path Constrained Viterbi Training Algorithm, Li et al (2000)[5], Ma et al (2009)[6], which restricts the possibilities of diagonal strings of states to propose a labeling, and updates all parameters in a pseudo-Expectation Maximization way using such labeling until convergence. We will describe now the equations involved in the process.

- **Choosing the best N paths for decoding** Path Constrained Viterbi has been introduced by Li et al (2000) [5], but they did not give any particulars

on how to choose these N sequences but the first one. Ma et al. (2009) [6] also worked with similar algorithms, keeping this crucial step also hidden as a part of their implementation.
In this paper, we propose the following selection. We firstly assume, as Li et al (2000) [5], that we can evaluate the likelihood of a given diagonal state sequence by simply multiplying the likelihoods of each pixel without considering statistical dependencies between pixels, i.e. we compute

$$\hat{p}(s_{ij} = \ell | I_{ij}, \hat{\theta}) \propto p(I_{ij} | s_{ij} = \ell, \hat{\theta}) \hat{p}(s_{ij} = \ell | \hat{\theta}) \tag{10}$$

where $p(I_{ij}|s_{ij}, \hat{\theta})$ is given by (2) and

$$\hat{p}(s_{ij} = \ell | \hat{\theta}) = \frac{\sum_{i=0}^{z-1} \sum_{j=0}^{w-1} \chi(s_{ij} = \ell)}{zw} \qquad \forall \ell \in \mathcal{L}.$$

Thus, the most likely state sequence $\mathbf{s}_{d,1}$ is the one that has in each entry the most likely state for the pixel's observation on diagonal $d \in \{0, 1, \ldots, z + w - 2\}$.
In our particular implementation, we will obtain the next $N - 1$ sequences considering only the sequences that result from changing only one state of $\mathbf{s}_{d,1}$. Such chains are ordered using (10) and the $N - 1$ with the largest likelihood are chosen.
In our Discussion section we will comment the incidence of the selection of this bag of N sequences in the convergence of our implementation.

– **Viterbi decoding over the chosen N paths.** We call each diagonal state sequence $\mathbf{s}_{d,k}$ where d is the index for the diagonal with $d = 0, 1, \ldots, z+w-2$ and $k = 1, 2, \ldots, N$ indicates the state sequence. Hence the initial state probabilities $\tilde{\pi}_k$ for pixel $(0, 0)$ are

$$\tilde{\pi}_k = p(T_0 = \mathbf{s}_{0,k}).$$

We denote $\delta_d(k)$ as the maximum probability for sequence k on diagonal d. Given the parameters of the PCVT we can write

$$\delta_d(k) = \max_{k_0, k_1, \ldots, k_{d-1}} p(\mathbf{s}_{0,k_0}, \ldots, \mathbf{s}_{d-1,k_{d-1}}, \mathbf{s}_{d,k}, \mathbf{I}_0, \ldots, \mathbf{I}_d | \theta),$$

with $d = 0, \ldots, z + w - 2; k = 1, \ldots, N$. Furthermore we collect the pixels on diagonal d in a variable $\Delta(d)$ and define

$$b_{\mathbf{s}_{d,k}}(\mathbf{I}_d) = \prod_{(i,j) \in \Delta(d)} p(I_{ij} | \mathbf{s}_{d,k}(i, j))$$

where $\mathbf{I}_d = (I_{ij} : (i, j) \in \Delta(d))$ and $b_{\mathbf{s}_{d,k}}(\mathbf{I}_d)$ is the emission probability of sequence k on diagonal d under the assumption that each pixel is statistically independent from its neighbors. Finally, we can calculate the transition probability from sequence k on diagonal d to sequence l on diagonal $d + 1$:

$$\tilde{a}_{d,k,l} = p(T_{d+1} = \mathbf{s}_{d+1,l} | T_d = \mathbf{s}_{d,k}, \theta) = \prod_{(i,j) \in \Delta(d+1)} a_{\mathbf{s}_{d,k}(i-1,j), \mathbf{s}_{d,k}(i,j-1), \mathbf{s}_{d+1,l}(i,j)}$$

$$d = 0, \ldots, z + w - 3; \quad k, l = 1, \ldots, N.$$

Now we are ready to initialize the *Viterbi decoding Algorithm* with the values

$$\delta_0(k) = p(T_0 = \mathbf{s}_{0,k}), \quad b_{\mathbf{s}_{0,k}}(\mathbf{I}_0) = \tilde{\pi}_k b_{\mathbf{s}_{0,k}}(I_{0,0}) \qquad \forall k = 1, 2, \ldots, N.$$

Then we start the recursion

$$\delta_{d+1}(l) = \left[\max_{1 \le k \le N} \delta_d(k)\tilde{a}_{d,k,l}\right] b_{\mathbf{s}_{d+1,l}}(\mathbf{I}_{d+1}) \qquad \forall d = 0, 1, \ldots, z+w-3 \quad \forall l = 1, 2, \ldots, N.$$

After each step, we save the index of the most probable sequence on diagonal d that leads to sequence l on diagonal $d + 1$ in a variable called φ:

$$\varphi_{d+1}(l) = \arg \max_{1 \le k \le N} \{\delta_d(k)\tilde{a}_{d,k,l}\} \qquad \forall d = 0, 1, \ldots, z+w-3 \quad \forall l = 1, 2, \ldots, N$$

When the algorithm reaches the last diagonal, we use the values saved in φ to track back the most probable path through the image starting with the bottom-right pixel

$$s^*_{z+w-2} = \arg \max_{1 \le k \le N} \delta_{z+w-2}(k)$$

$$s^*_d = \varphi_{d+1}(s^*_{d+1}) \qquad \forall d = z + w - 3, z + w - 4, \ldots, 1$$

The final result s^* contains the optimal path through the N sequences at each diagonal. Note that this is equal to knowing the complete hidden state map for the whole image.

4 Experimental Results: Image Classification

In this section we report some experiments on the algorithms described in this article: Path Contrained Viterbi Training (PCVT) and Iterated Conditional Modes (ICM).For comparison purposes, we also provide the results when applying supervised (ML) or unsupervised (EM-ML) Maximum Likelihood Classification.

4.1 Multiclass high-quality bitewing X-ray image.

Digitalized X-ray images have some level of noise introduced by the scanner, but their main characteristic is the smoothness of the joint gray level histogram. Classes that are quite distinguishable to the naked eye do not form a distinctive mode in the joint histogram, making segmentation difficult. Image subtraction, image enhancement and filtering are common image processing research areas when working with digitalized or digital X-ray imagery. Caution has been advised by dentists [8] about the abuse of enhancement algorithms in digital X-ray devises, that often introduce artifacts in the images by defect and by excess, leading to possible misdiagnosis.

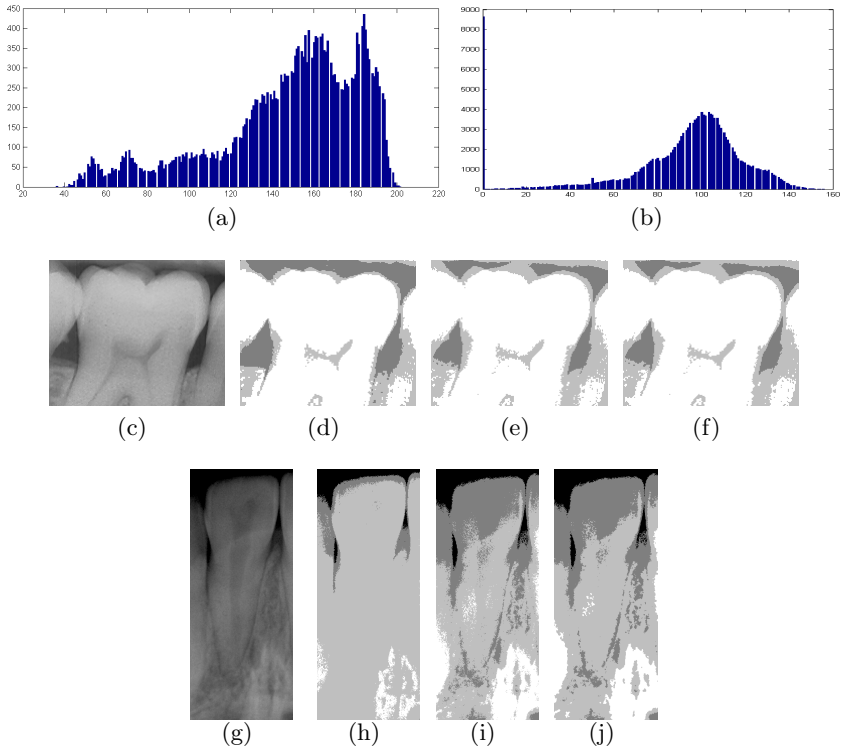

Fig. 2. Segmentation of inverse digitalized dental X-ray images. Panels (a) and (b) histograms of pixel intensities corresponding to molar image (c) and incisive image (g). PCVT segmentations are shown in panel (d) and (h), ML in panels (e) and (i) and ICM in panels (f) and (j).

Both images have a central tooth and partial views of its neighbor teeth, gums and background. We have set three classes for the first image and four for the second image to account for the differences between tooth enamel and dentin. Enamel is the thin, hard material that covers the dentin, or main body of the teeth, and protects it from harsh temperatures. We initialized both algorithms with supervised classification. In Figure 2 we can observe from the images that, in the first case, the teeth are correctly segmented, dentine is clearly differentiated from tooth nerve and enamel. The second image has a flat histogram, classes are mixed together and segmentation is more difficult. All algorithms perform badly in this case.

4.2 Simulated binary images and Kappa statistic

In this section we want to discuss the influence of the selection of the best N sequences for decoding in execution time and overall performance. We de-

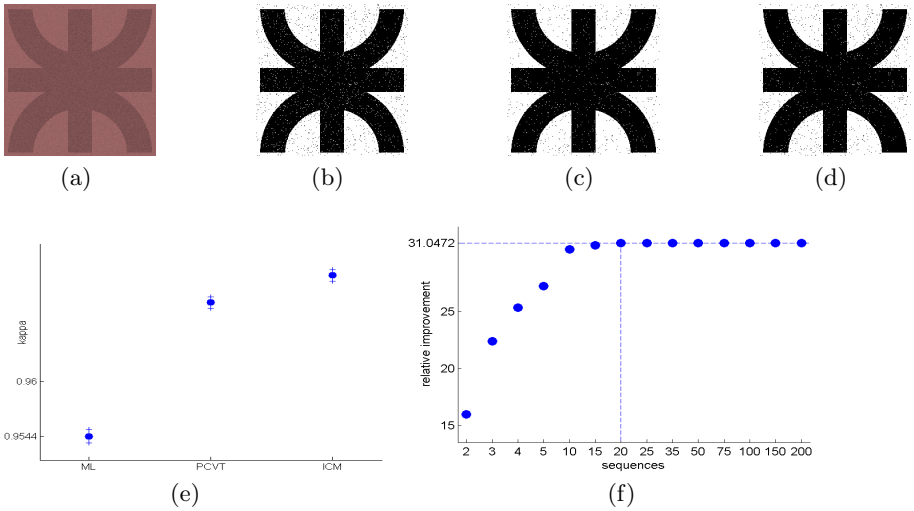

Fig. 3. (a) Noisy UTN logo, (b)EM-ML segmentation, (c)EM-ICM segmentation, (d) EM-PCVT segmentation, (e) confidence intervals for kappa statistic, (f) Relative improvement of PCVT related to ML vs number of sequences retained. Remaining noise is concentrated on the background for ICM and PCVT, while ML has misclassified pixels in both classes.

vised an unsupervised study with the 2-color logo of the Technological University degraded with gaussian noise. For this image, the confidence intervals for Kappa show that PCVT and ICM are significantly different from ML, see panel (c) of Figure 3. The index Kappa is defined by$\kappa = \frac{P_o - P_e}{1 - P_e}$ where P_0 is the observed proportion of agreement and P_e is the expected proportion of agreement in the image, one computed over the estimated segmentation map B and the other computed over the ground truth map V. The OA overall accuracy statistic is the number of well classified pixels over the total number of pixels. When appropriate, we also report the Relative Improvement index related to the benchmark ML or EM-ML classification, defined as $Relative\ Improvement = 100 \times (OA_{method} - OA_{ML})/(100 - OA_{ML})$.

Now we discuss the number of sequences N retained for decoding. Our personal implementation allows the user to set the number of sequences involved in the search, being 250 the preset value. We made 16 experiments setting the number of path sequences allowed for decoding in a range from 1 to 250, as we can see in panel (f) of Figure 3. We also computed time until convergence, number of iterations until convergence and relative improvement of classification accuracy related to EM-ML segmentation. This study shows that allowing the most probable 20 sequences has the same relative improvement as working with the most probable 250, and the time of execution goes from 0.8 minutes to 59 minutes on an Intel I7 processor, 6Gb memory HP laptop. The number of iterations stabilizes when 50 or more sequences are allowed.

5 Conclusion

In this paper we revisited two different Markovian models and its most noticeable estimation algorithms. The complexity of the algorithms is quit different, since Pott's ICM has only one parameter to set and PCVT has all transitions probabilities to estimate besides the Gaussian parameters. Nevertheless, in our initial study we found PCVT segmentations promising. ICM has a tendency to smooth out the initial ML segmentation. PCVT has the capability of moving out from the saddle point where ML lays in the space of all possible segmentations and deliver a different segmentation. This is important in the case of images with several mixed classes.

The PCVA code we made to carry out these experiments was written from scratch, on a Matlab 2013a platform. We used Matlab Statistical toolbox scripts for (EM-ML) and (ML). In the literature, initialization has also been made with non-parametric segmentation algorithms like k-means, while the means and variances for the Gaussian hypothesis on the observations were estimated over the labeled output. For the studied examples, we did not observe significant differences using k-means as initialization method. We implemented a version of ICM where parameter β is estimated at each iteration by maximizing the current pseudo-likelihood with the Brent algorithm. We will continue working with more challenging types of images, introducing also other measures besides Kappa to study performance. All code written is provided in a Matlab toolbox available for download from our website, following the Reproducible Research Paradigm.

References

1. S. Y. Chen, Hanyang Tong, and Carlo Cattani (2012) Markov Models for Image Labeling, Mathematical Problems in Engineering, vol. 2012, Article ID 814356, 18 pages, 2012.
2. J. Gimenez, A. Frery, A. G. Flesia (2013) Inference strategies for Potts model. To appear Procc. IGARSS 2013.
3. A. C. Frery and S. Ferrero and O. H. Bustos (2009)The Influence of Training Errors, Context and Number of Bands in the Accuracy of Image Classification, International Journal of Remote Sensing, 30:6, 1425–1440.
4. A. Levada, N. Mascarenhas, A. Tannus (2010) A novel MAP-MRF approach for multispectral image contextual classification using a combination of suboptimal iterative algorithms *Pattern Recognition Letters*: 31 :1795–1808.
5. J. Li, A. Najmi, and R. M. Gray (2000) Image classification by a two-dimensional Hidden Markov model, IEEE Transactions on Signal Processing, vol. 48, no. 2, pp. 517–533, Feb. 2000.
6. Xiang Ma, D. Schonfeld and A. Khokhar. (2009). Video event classification and image segmentation based on noncausal multidimensional hidden markov models. IEEE transactions on image processing, vol 18, N 6, pp 1304–1313.
7. M. E. Sargin, A. Altinok, K. Rose, B. S. Manjunath (2008) Conditional iterative decoding of two dimensional Hidden Markov models. Proceedings ICIP 2008.
8. J G Flesia and A G Flesia (2011) The influence of processing in the accuracy of measurements in indirect digitalized intra-oral radiographic imaging for forensic applications. The Forensic Oral Pathology Journal,vol 2, N4, pp. 20–24.