Edge modelling MRFs for cardiac MRI segmentation

Janto Dreijer

***Abstract*—This paper presents a simple template for IEEEtran documents.**

***Index Terms*—simplicity, beauty, elegance**

I. Introduction

MAGNETIC Resonance Imaging (MRI) provides images with high contrast between water-containing organs and their surrounding tissue. Multiple frames can be captured in succession to create a video of organs at a specific spatial slice. This makes it useful for diagnosing heart related problems. Segmentation of the heart is important in calculating properties such as ejection rate and epicardium thickness. Segmentation is a labor intensive task, however, and various automated cardiac segmentation algorithms have been devised.

Markov Random Fields (MRFs) are a popular technique in medical segmentation. Segmentation is posed as a problem of associating a label (often binary) to regularly spaced sites (often per pixel). MRFs then model the relationship between a site's label and the observed pixel value at a site as well as a site's label and its neighbor's label. Efficient inference is often done through belief propagation or graph cuts. This works especially well when extracting homogenous objects such as the blood pool. The left ventricle, however, contains two papillary muscles. When modelling the cell wall it is desirable to include these muscles in the segmentation. The graphical model for this MRF effectively creates a surface of nodes over the image. We therefore refer refer to these techniques as surface MRFs. This type of model does not encode any shape information. Brain models?

Information on the shape of the cardiac wall and surrounding structures is typically exploited with the use of Active Shape Models (ASMs). This transforms the segmentation's contour into a shape space in which searches are done. Integrating temporal information, however, is only often implicitly through an online tracking algorithm (e.g. greedy or Kalman filter).

We choose to model the contour edge directly in what we call an edge MRF. This allows us to integrate shape and temporal information explicitly through feature functions. By applying belief propagation for inference this naturally leads to offline segmentation.

We will describe our edge model in more detail, briefly describe MRFs and how inference and training is done. We concluded with quantitative and qualitative evaluation and a comparison to existing methods followed by concluding remarks.

Your name is with Stellenbosch University

II. Related Work

Andreopoulos and Tsotsos [1] fit a 3D active appearance model and investigate a hierarchical 2D + time active shape model (ASMs), that integrates temporal constraints by using the third dimension for time instead of a spatial dimension.

Various heuristic methods exist to segment cardiac MRI images. One that is of interest due to its similarity in configuration space is... It is interesting to compare ... They also work in logpolar space in which the least cost path from top to bottom is defined as the wanted contour. Defining cost function to realize this is neccesary. If we limit our model to a single frame (i.e. remove temporal linkage) belief propagation will find an estimate of the least cost path through a cost function defined on the polar image.

III. Problem Formulation

Problem statement: when given a single frame $\boldsymbol{D}(t) = g(x, y)$ at time t we wish to find the contour $\boldsymbol{z}(t) = (\boldsymbol{x}(t), \boldsymbol{y}(t))$ around the endocardium border.

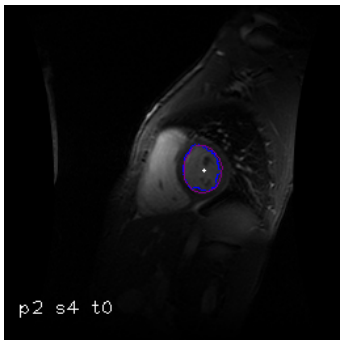

Figure 1. Captions go *under* the figure

Because we are working with spherically shaped contours we can simplify the configuration space by choosing a center point $c(t)$ for each frame and working in the logpolar transformed version of the variables. $z_n = (x_n, y_n) = (r_{\text{init}} \exp(\rho_n), \phi_n) + c$.

This reduces the size of the configuration space of nodes. Except for computational efficiency this also has the effect of making all points within the state space valid realistic contours. As apposed to the nonsensical shapes possible if we e.g. connect random points in cartesian space.

$\rho_n(t)$

$\boldsymbol{D}_n(t)$

By knowing the center point we can then refer to the configuration of a single frame as $\boldsymbol{\rho}(t)$ (i.e. a sequences of radii).

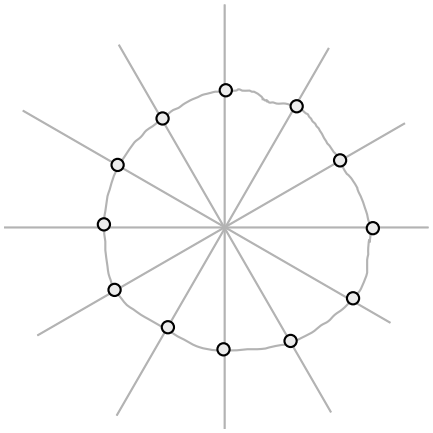

Figure 2. Coordinates on contour

When we are provided with a series of successive frames we would like to leverage temporal information to find a series of contours. We therefore generalize our notation to represent videos and their segmentations: $\boldsymbol{D} = \{\boldsymbol{D}(t),\, t \in [1..T]\}$, $\boldsymbol{z} = \{\boldsymbol{z}(t),\, t \in [1..T]\}$, $\boldsymbol{\rho} = \{\boldsymbol{\rho}(t),\, t \in [1..T]\}$, $\boldsymbol{\rho} = \{\boldsymbol{\rho}(t),\, t \in [1..T]\}$, $\boldsymbol{c} = \{c(t),\, t \in [1..T]\}$.

We simplify our problem further with a preprocessing step that attempts to align frames in a video so they share a common center point. We assume an adequate value for $c(0)$, the center point for the first frame, is provided by the user. Many heuristic techniques have also been devised to find a rough estimate of the endocardium's location. The preprocessing tries to find a good center point for the next frame by minimizing the error between aligned frames. Errors are weighted normally from the previous frames center point. A side effect is that we have eliminated translations from our data and allowed our imodel to focus only on deformation of the size and shape. As long as $c(0)$ lies within the endocardium we have observed good results. Alternating between inferencing $\boldsymbol{\rho}$ and re-estimating c is also possible and we believe will lead to improved results.

IV. Markov Random Field

A probabilistic formulation of our problem allows us to apply various modeling and inference techniques. Specifically if we are given a sequance of frames $\boldsymbol{D}$ we would like to find the set of contours $\boldsymbol{\rho}^{\star}$ that maximizes the conditional $P(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D})$. $\boldsymbol{\theta}$ is a set of parameters to be learnt.

We model the relationship between our variables ($\boldsymbol{\rho}$, $\boldsymbol{D}$ and $\boldsymbol{\theta}$) with a Markov Random Field G. An MRF expresses conditional independance between random variables and can be used as a basis for inferencing the most likely values of unknown variables. We use the class of MRFs that can be factorized in terms of its cliques with a partition function Z that normalises the product into a p.d.f. A single slice of our MRF is shown in Figure 3. Only $N = 6$ is shown and in reality T rings are connected as shown in Figure 4.

Through the Hammersley-Clifford theorem we can express the conditional in terms of the product of clique potential functions

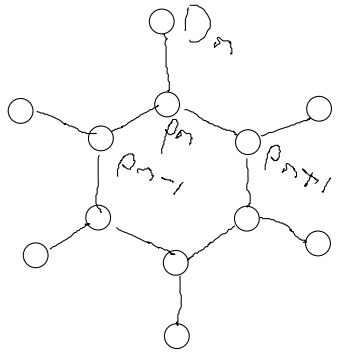

Figure 3. Graphical model of a single slice

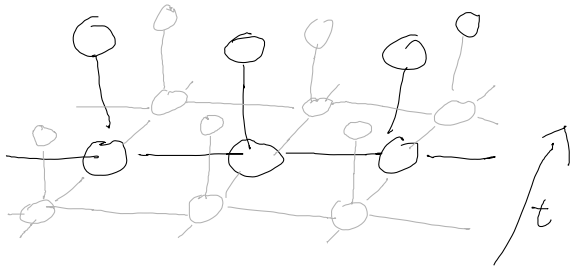

Figure 4. Graphical model with temporal info

$$P(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D}) = \frac{1}{Z(\boldsymbol{\theta})} \prod_{C \in \mathrm{cl}(G)} \psi_C(\boldsymbol{\rho}_C|\boldsymbol{\theta}_C, \boldsymbol{D}_C) \tag{1}$$

with a partition function $Z(\boldsymbol{\theta})$ that normalizes the product.

$$Z(\boldsymbol{\theta}) = \sum_{\boldsymbol{\rho}} \prod_{C \in \mathrm{cl}(G)} \psi_C(\boldsymbol{\rho}_C|\boldsymbol{\theta}_C, \boldsymbol{D}_C) \tag{2}$$

Expressing all potential functions as a log linear combination of weighted feature functions $\psi_C = \exp(\theta_C f_C(\boldsymbol{\rho}_C, \boldsymbol{D}_C))$ allows us to simplify the above into

$$P(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D}) = \frac{1}{Z(\boldsymbol{\theta})} \exp(-E(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D})) \tag{3}$$

With energy defined as

$$E(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D}) = \sum_{C} \theta_C f_C(\boldsymbol{\rho}_C, \boldsymbol{D}_C) \tag{4}$$

and partition function

$$Z(\boldsymbol{\theta}) = \sum_{\boldsymbol{\rho}} \exp(-E(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D})) \tag{5}$$

Note that feature functions are defined over the variables in a clique.

Expressing the probability of a configuration in terms of an energy function simplifies many calculations.

A. Feature Functions

We encode our dependancies as a weighted sum of feature functions. Functions were experimentally designed to yield small values for our segmentation and larger ones otherwise. Feature functions are positive and normalised to mostly be smaller than 1. This allows us to take $\boldsymbol{\theta}$ as positive and interpret it as purely scaling values.

Three types of cliques (and thus feature functions) exist based on the types of joint probabilities modeled: image-node dependance $\psi_c\left(\rho_n\left(t\right), D_n\left(t\right)\right)$, spatial node-node dependence $\psi_c\left(\rho_n\left(t\right), \rho_{n-1}\left(t\right)\right)$ and temporal node-node dependance $\psi_c\left(\rho_n\left(t\right), \rho_n\left(t-1\right)\right)$.

1) image-node functions: We seek a small response for strong negative edges (edge lies from light to dark areas going radially outwards)

$$f_1\left(\rho_n(t), D_n(t)\right) = 1 + \left(D_{n+1}(t) - D_n(t)\right)/255 \quad (6)$$

$$\text{mean}(n,t,m) = \frac{1}{m}\sum_{m'} D_n(t,m') \quad (7)$$

minimize included deviation from mean

$$\text{var}(n,t,m) = \frac{1}{m}\sum_{m'}\left(D_n(t,m') - \text{mean}(n,t,m)\right)^2 \quad (8)$$

$$f_2\left(\rho_n(t), D_n(t)\right) = \sqrt{\text{var}(n,t,m)} \quad (9)$$

color ratio. less sensitive to scaling of pixel values

$$f_3\left(\rho_n(t), D_n(t)\right) = \left(\arctan\frac{D_n}{D_{n+1}}/\left(2\pi\right)\right)^2 \quad (10)$$

2) spatial node-node functions: Through analysis of shapes in training sets we see that segmentations are elliptical with vertical axis longer than horizontal. The difference between neighboring radii is thus dependance on their angular position. Specifically μ is a function of n:

$$\rho_n\left(t\right) - \rho_{n-1}\left(t\right) \sim \mathcal{N}\left(\mu\left(n\right), \sigma^2\left(n\right)\right) \quad (11)$$

We therefore introduce a feature function as follows:

$$f_2^{\text{reach}}\left(\rho_n(t), \rho_{n-1}(t)\right) = \left(\left(\rho_n(t) - \rho_{n-1}(t)\right) - \mu\left(n\right)\right)^2/M^2 \quad (12)$$

We also try to minimize the color difference between nodes

$$f_2^{\text{reach}}\left(\rho_n(t), \rho_{n-1}(t)\right) = \left(D_n(t,\rho_n(t)) - D_{n-1}(t,\rho_{n-1}(t))\right)^2/255^2 \quad (13)$$

3) temporal node-node functions: There is definate shrinkage over sistole and growing over diastole, however we cannot depend on a series of frames starting and ending at peak diastole.

$$\rho_n\left(t\right) - \rho_n\left(t-1\right) \sim \mathcal{N}\left(\mu\left(t\right), \sigma^2\left(t\right)\right) \quad (14)$$

Therefore we assume edge coordinates changes gradually over time $\mu\left(t\right) = 0$

$$f_1^{\text{time}}\left(\rho_n(t), \rho_n(t-1)\right) = \left(\rho_n(t) - \rho_n(t-1)\right)^2/M^2 \quad (15)$$

V. Inference

When a segmentation is required for a provided sequence of frames we run an inference algorithm on our MRF. Specifically we want to find the configuration $\boldsymbol{\rho}^\star$:

$$\boldsymbol{\rho}^\star = \arg\max_{\boldsymbol{\rho}} P\left(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D}\right) \quad (16)$$

However, because $Z\left(\boldsymbol{\theta}\right)$ is independant of $\boldsymbol{\rho}$, we can reduced this to finding

$$\boldsymbol{\rho}^\star = \arg\min_{\boldsymbol{\rho}} E\left(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D}\right) \quad (17)$$

For inference problems that can be expressed as a tree we can run the min-sum algorithm and calculate an exact value for $\boldsymbol{\rho}^\star$. Unfortunately because of the linkage between frames and the spatial circular connectivity of each frame we have a graphical model with many loops. There is currently no known way to efficiently calculate $\boldsymbol{\rho}^\star$ exactly as maximization of the above is NP-complete, however loopy belief propagation has been used with success in the past. Sufficient conditions for convergence are coming to light.

A. Factor Graphs

Factor graphs are bipartite graphs that indicate the relationship of feature functions (factor nodes) and the variables which they are taken as arguments (variable nodes). This is not only useful as a design tool, but also simplifies illustration of belief propagation calculations. Our factor graph is shown in Figure 5.

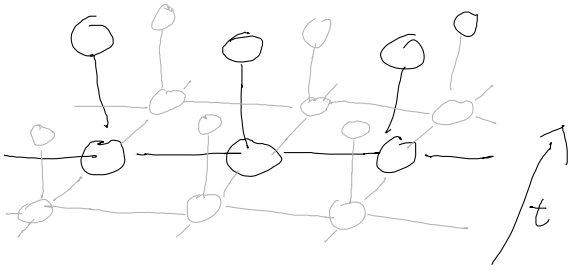

Figure 5. Factor graph

In belief propagation messages sent from a factor node summarizes away lower nodes by maximizing over lower variables. Our implementation only stores messages emmited from factor nodes as those sent from variable nodes are already

included in their calculation. The node values that maximizes parts in a message is also stored for use when backtracking.

Belief propagation requires one to specify a specific shedule by which messages are passed through the graph. We initialise all messages to zero and propagate through one frame at a time. We pass messages sequentially through nodes in each frame.

Messages $m_{\mathrm{L}}\left(\rho_{n-1}(t)\right)$ is an array of length M, i.e. a value evaluated for each value $\rho_n(t)$

$$m_{\mathrm{L}}\left(\rho_{n-1}(t)\right) = \min_{\rho_n(T)} \sum_k \theta_k f_k\left(\rho_n(t), \rho_{n-1}(t)\right) + m_{\mathrm{L}}\left(\rho_n(t)\right) + m_{\mathrm{up}}\left(\rho_n(t-1)\right) + m_{\mathrm{up}}\left(\rho_n(t+1)\right)$$

Through experimentation # interations seem to work best. We propagate messages within a frame before propagating them to the next frame.

Additionally we implement our belief propagation with a beam search. For example, evalutation over all ρ_{n-1} when calculating $\max_{\rho_{n-1}} m_{\mathrm{R}}(\rho_n, \rho_{n-1})$ is often unnecesary due to spatial continuity restrictions in our contours. Specifically through inspection we found that $|\rho_n(t) - \rho_{n-1}(t)| < 6$ and $|\rho_n(t) - \rho_n(t-1)| < 50$ in our training set.

$$\rho_n(t) = \arg\max_{\rho_n(t)} P(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D}) \tag{18}$$

$$\rho_N(T)^\star = \arg\max_{\rho_N(T)} P(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D}) \tag{19}$$

$$S(\rho_N(T)) = m_L\left(\rho_N(T)\right) + m_R\left(\rho_N(T)\right) + m_{\mathrm{up}}\left(\rho_N(T)\right)$$

$$\rho_N(T)^\star = \arg\min_{\rho_N(T)} S(\rho_N(T)) \tag{20}$$

Backtracking in frame T. First find $\rho_N(T)^\star$. Find the $\rho_{N-1}(T)$ that maximizes $m\left(\rho_N(T)\right)$. this is repeated for all N nodes in frame T. We can then move to the next frame $T-1$ by finding the value $\rho_n(T-1)$ that maximized $m_{\mathrm{up}}(\rho_n(T)^\star)$. This is done for all N and we then continue to the next frame.

VI. Training

Finding the best parameters, $\boldsymbol{\theta}^\star$, is done by considering a set of training videos, $\boldsymbol{D}^{\text{train}}$, and their provided segmentations, $\boldsymbol{z}^{\text{train}}$.

Maximum likelihood estimation often works well for other kinds of machine learning training problems, so it's natural to attempt a similar formulation. However we can see from above that $Z\left(\boldsymbol{\theta}\right)$ would require evaluation which is mathematically intractible (belief prop?). Various ways to address this difficulty exist such as pseudo likelihoods and monte carle techniques.

We attempt to avoid this issue by rather searching for the value for $\boldsymbol{\theta}$ that would minimize the error between segmentation that is yielded by the inference process and the segmentation in the training set.

$$\boldsymbol{\theta}^\star = \arg\min_{\boldsymbol{\theta}} \operatorname{dist}(\boldsymbol{\rho}^{\text{train}}, \boldsymbol{\rho}^\star) \tag{21}$$

$$\boldsymbol{\rho}^* = \arg\max_{\boldsymbol{\rho}} P\left(\boldsymbol{\rho}|\boldsymbol{\theta}, \boldsymbol{D}\right) \tag{22}$$

Because we can calculate the above (inference) without having to calculate $Z\left(\boldsymbol{\theta}\right)$ we avoid this issue. This does leave us in a position where we are not able to directly calculate a gradient in $\boldsymbol{\theta}$-space that would be useful in iterative techniques.

Iterative techniques that estimate local gradients (such as BFGS) could potentially be used. However there is the problem of local optima and saddle points. When working in higher dimensions estimating the gradient is expensive and may be of a diminishing return.

We employ random search technique: From an initial guess for $\boldsymbol{\theta}$ we sample from a normal distribution centered at this value $\boldsymbol{\theta}_\tau \sim \mathcal{N}\left(\boldsymbol{\theta}_{\tau-1}, \sigma^2\left(\tau\right)\right)$, shrinking the variance $\sigma^2\left(\tau\right)$ over iterations. If the sampled $\boldsymbol{\theta}$ produces a smaller error we use this one in the next iteration.

This is similar to simulated annealing, however we do not have an acceptance test and only favour moves that go "downhill". This adds to the risk of not being able to escape from local minima.

convergence

VII. Evaluation

We evaluate our methodology against the York dataset [1]. Originally developed to showcase ASM tracking it contains 30 patients each with around 10 sequential frames at around 10 different slice positions. Each frame is annotated with the endocardium border. Videos of 15 patients are taken for training and 15 for testing.

It currently takes around 2 seconds per frame to run inference using a 2.6 GHz processor.

numbers

runtime

Table I
TABLE CAPTIONS GO *above* THE TABLE

delete	this
example	table

VIII. Conclusions

We hope further techniques become available that will allow offline tracking to be prefered choice. Offline tracking should be more robust and not lose tracking as easily.

Shape information is local as apposed to the global nature of the state space in which ASMs work. This seems to place some limitations on MRF's expressive power.

It is easy to integrate temporal information into an MRF. The probabilistic nature allows us to gradually integrate additional information as it becomes necessary.

We hope to have demonstrated through our application of edge MRFs to cardiac segmentation that edge similar techniques could hold promise for segmentation and tracking in general.

Your Name All about you and the what your interests are. Don't forget to put your name in between a pair of {}'s that are set as raw TEX.

Coauthor Same again for the co-author.

REFERENCES

[1] Alexander Andreopoulos and John K. Tsotsos. Efficient and generalizable statistical models of shape and appearance for analysis of cardiac mri. *Medical Image Analysis*, 12(3):335 – 357, 2008.