AN ANALYSIS OF THE MAP SEEKING CIRCUIT AND MONTE CARLO EXTENSIONS

Zeynep Engin, Jeffrey Ng, Mauricio Barahona, Anil A. Bharath

Imperial College London, Faculty of Engineering South Kensington Campus, London, SW7 2AZ, UK

ABSTRACT

The Map Seeking Circuit (MSC) has been suggested to address the inverse problem of transformation discovery as found in signal processing, vision, inverse kinematics and many other natural tasks. According to this idea, a parallel search in the transformation space of a high dimensional problem can be decomposed into parts efficiently using the ordering property of superpositions. Deterministic formulations of the circuit have been suggested. Here, we provide a probabilistic interpretation of the architecture whereby the superpositions of the circuit are seen as a series of marginalisations over parameters of the transform. Based on this, we interpret the weights of the MSC as importance weights. The latter suggests the incorporation of Monte-Carlo approaches in the MSC, providing improved resolution of parameter estimates within resource constrained implementations. As a final contribution, we model mixed serial/parallel search strategies of biological vision to reduce the problem of collusions, a common problem in the standard MSC approach.

Index Terms— map seeking circuit, pattern detection, transformation estimation, marginalisation

1. INTRODUCTION

The goal of a Map Seeking Circuit (MSC) is to find the best sequence of transformations that matches an input pattern to one or more stored templates [1]. The MSC architecture, which decouples the search of transformation parameters from each other, is a plausible mechanism to explain the speed of cortical networks. From a computational perspective, the superposition of hypothetical transformations, combining multiple transformations of the same type (e.g. translation, rotation, scaling), allows the effect of transformation types to be isolated from each other, and to evaluate error functionals over single transformation parameters. The generality of the method offers solutions to a wide range of problems including visual, other perceptual, cognitive, and motor applications. The underlying principles of the circuit are still being analysed. Recent work has described the architecture and its computational processes as a discrete dynamical system [2], and in terms of difference equations [3]. Our first contribution is to interpret the MSC in a more mainstream statistical framework; accordingly we proceed as follows: first, we formulate the MSC in a probabilistic framework, whereby the estimation of the transformation sequence is treated as a problem of estimating a joint distribution of transformation parameters. This provides probabilistic interpretations on the superposition of hypothesised transformations, a key feature of the standard MSC, and an understanding of the limitations of the components and operation of the standard MSC. Our second contribution stems directly from this: the formulation of the MSC in a probabilistic framework suggests the introduction of sequential importance resampling processes in adaptively exploring the transformation parameter spaces. This potentially achieves higher precision in estimates of the transformation parameters within roughly the same computational resources as the original MSC in our Monte Carlo Map Seeking Circuit (MC-MSC) scheme. Our third contribution addresses the problems introduced by superimpositions occurring in the translation layer of a cluttered scene; here, we turn to the biological strategy suggested by employing fixational eye movements and visual attention: essentially a mixture of sequential and parallel search strategies of biological visual systems [4]. We achieve this by introducing a queuing system into the MC-MSC architecture at the translational level; this dramatically reduces the effects of collusions at this level of the system.

2. MARGINALISATION TO PARTITION HIGH DIMENSIONAL SEARCH SPACES

Consider the problem of finding the best sequence of a given set of transformations that matches an input signal, I, to the stored template, M, through a series of comparisons between the transformed signals, templates and superpositions of those. Let $f_i, i \in N$ denote the i^{th} transformation parameter in the sequence. The correct sequence, $(f_1, ..., f_L)$ can be formulated in the Bayesian framework as

$$P(f_1, ..., f_L | I, M) = \frac{P(I, M | f_1, ..., f_L) P(f_1, ..., f_L)}{P(I, M)}$$
(1)

where assuming a multivariate conditional distribution (or any exponential form) for the likelihood, $P(I, M|f_1, ..., f_L)$, the search for the maximum location can be carried out with the log-likelihood instead. This model becomes computationally intractable for moderate to large numbers of parameters owing to the large number of combinations to be taken into account, leading to the problem of combinatorial explosion. Furthermore, estimating the likelihood for each combination involves transforming images prior to the comparison, adding to the computational load. The problem of estimating high dimensional joint likelihood density functions can be addressed either by assuming independence of the variables or by marginalisation. In this case, the independence assumption is not plausible because each transformation in the sequence affects the final image in a complex manner. Marginalisation captures the correlation between variables and could help to reduce the dimensionality of the search space. This equates to splitting the joint likelihood into a set of marginalised probability density functions each defined over only one variable f_i . To simplify the search, one would ideally marginalise over all parameters in the sequence except for one at a time. For L different variables, L - 1 marginalisations would be required. So, the collapsing of the search space into 1D over f_i can

Thanks to strategic initiative of Imperial College for partial funding.

be summarised by the following:

$$P(I, M|f_i) = \int \dots \int P(I, M|f_1, \dots, f_L) P(f_0) \dots$$

$$P(f_{i-1}) P(f_{i+1}) \dots P(f_L) df_1 \dots df_{i-1} df_{i+1} \dots df_L$$
(2)

which is usually computed by summing the conditional probabilities over all possible values of the transformation parameters. We suggest that the MSC of Arathorn may be interpreted as a series of marginalisations over components of a series of transformations. For example, to average out the effect of the first and last transfor-



Fig. 1. An intermediate layer of the MSC

mations, we can perform weighted superpositions of the transformed signal and the template respectively. In the generic case, any parameter can either be averaged out by removing the previous or following transformations from the signal and the template respectively, depending on how close they are in the sequence [1]. Thus, the MSC uses the superposition of the marginalised transformations, I_{i-1} , instead of the signal, I, and the superpositions of the backwards transformations, M_{i+1} , instead of the template, M (A typical layer for the search on i^{th} parameter in the sequence is seen in Fig.1). The posterior density function for one variable is then defined as

$$P(f_i|I_{i-1}, M_{i+1}) = \frac{P(I_{i-1}, M_{i+1}|f_i)P(f_i)}{P(I_{i-1}, M_{i+1})}$$
(3)

where $P(f_i|I_{i-1}, M_{i+1}) = P(f_i|f_1, ..., f_{i-1}, f_{i+1}, ..., f_L, I, M)$ is implicit and the likelihood is computed by

$$P(I_{i-1}, M_{i+1}|f_i) = \int P(I_{i-1}, M_{i+1}|f_{i-1}, f_i) P(f_{i-1}) df_{i-1}$$
(4)

Let N_i be the number of different transformations for the i^{th} parameter in the sequence. The number of operations needed to search the whole space would reduce from $\prod_i^L N_i$ to $\sum_i^L N_i$ with this process. Although the search over marginalised spaces is still in nature combinatorial, the number of computations is reduced considerably giving an opportunity to attack many transformation estimation problems.

3. MC-MSC DESCRIPTION FOR TWO LAYERS

A Monte Carlo approach allows dynamical evaluation of likelihood distributions in time [5]. The search for the maximum in a highly complex likelihood function is in this case performed through a set of marginal likelihoods instead of the joint distribution in an iterative process that we call Monte Carlo Map Seeking Circuit (MC-MSC). For relative simplicity, we illustrate the process of the MC-MSC for a two layer case in which rotation and scaling are to be

infered between an image, I, and a stored template, M (In practice, we also have layers for translation). We start by initialising the parameters uniformly over the search domain, $s^{(0)} \sim U(a,b)$, $\theta^{(0)} \sim U[0,2\pi)$, and the scaling parameters having probabilities inversely proportional to themselves, $p(s^{(0)}) = 1/s^{(0)}$ (note that $P(\theta^{(0)})$ is not used in the backwards path in two layer case). For iteration $k~(k\geq 1)$, the backward path is run first to obtain hypotheses of what the template will look like through all possible transformations considered at each layer. Only the second layer hypothesis is needed for a two layer circuit. Thus, samples of inversely scaled templates, $M_s^{(k-1)} = T_s^{-1} \{M\}(s^{(k-1)})$ are drawn as $M_s^{(k-1)} \sim p(M_s^{(k-1)}|s^{(k-1)})p(s^{(k-1)})$ and $p(M_s|s) = N(T_s\{M\}, \sigma_M^2; H)$. The expected value of $M_s^{(k-1)}$ is obtained by marginalising over the scaling parameter, $s^{(k-1)}$ as

$$E(M_s^{(k-1)}) = \overline{M}_s^{(k-1)} = \int M_s^{(k-1)} p(s^{(k-1)}) ds^{(k-1)}$$
(5)

The new distributions based on the backward information are chosen as $s^{(k)} \sim \mathcal{N}(s^{(k-1)}, \sigma_s^2)$, $p(s^{(k)}) = 1/s^{(k)}$, and $\theta^{(k)} \sim \mathcal{N}(\theta^{(k-1)}, \sigma_{\theta}^2)$. Then, we draw samples from the rotated input signals, $I_{\theta}^{(k)} = T_{\theta}\{I\}(\theta^{(k)})$ as $I_{\theta}^{(k)} \sim p(I_{\theta}^{(k)}|\theta^{(k)})p(\theta^{(k)}) p(I_{\theta}|\theta) = \mathcal{N}(T_{\theta}\{I\}, \sigma_I^2; H)$. The likelihood of each of these model hypothesis is computed by comparison with the imagery hypothesis at that layer (fed as an input from the next layer in the architecture), $M_s^{(k-1)}$, as follows

$$p(I_{\theta}^{(k)}|\theta^{(k)}, \overline{M}_{s}^{(k-1)}, \Sigma_{\theta s}; H) = \frac{1}{(2\pi)^{N/2} |\Sigma_{\theta s}|^{(1/2)}} \times \exp\left(-\frac{1}{2} (I_{\theta}^{(k)} - \overline{M}_{s}^{(k-1)})^{T} \Sigma_{\theta s}^{-1} (I_{\theta}^{(k)} - \overline{M}_{s}^{(k-1)})\right)$$
(6)

and the inference on θ at this layer is obtained by Bayes' theorem as

$$p(\theta^{(k)}|I_{\theta}^{(k)}, \overline{M}_{s}^{(k-1)}, \Sigma_{\theta s}^{(k)}; H) = \frac{p(I_{\theta}^{(k)}|\theta^{(k)}, \overline{M}_{s}^{(k-1)}, \Sigma_{\theta s}^{(k)}; H)p(\theta^{(k)})}{\int p(I_{\theta}^{(k)}|\theta^{(k)}, \overline{M}_{s}^{(k-1)}, \Sigma_{\theta s}^{(k)}; H)p(\theta^{(k)})d\theta(k)}$$
(7)

The mean \overline{I}_{θ} , according to the weighting distribution, $p(\theta^{(k)})$ is again computed as

$$E(I_{\theta}^{(k)}) = \overline{I}_{\theta}^{(k)} = \int I_{\theta}^{(k)} p(\theta^{(k)}) d\theta$$
(8)

In the next layer, the scaled samples of the input $I_s^{(k)} = T_s\{\overline{I}_{\theta}\}(s^{(k)})$ are drawn from $I_s^{(k)} \sim p(I_s^{(k)}|s^{(k)},\overline{I}_{\theta}^{(k)})p(s^{(k)})$ and $p(I_s|s,\overline{I}_{\theta}^{(k)}) = \mathcal{N}(T_s\{\overline{I}_{\theta}\},\sigma_1^2;H)$ then the likelihood of each of these samples is again obtained by a comparison between these samples and the template directly

$$p(I_s^{(k)}|M, \Sigma_{s\theta}; H) = \frac{1}{(2\pi)^{N/2} |\Sigma_{s\theta}|^{(1/2)}} \times \exp\left(-\frac{1}{2} (I_s^{(k)} - M)^T \Sigma_{s\theta}^{-1} (I_s^{(k)} - M)\right)$$
(9)

These likelihoods are used to construct an estimate of the distribution of the scaling parameter, $s^{\left(k\right)}$

$$p(s^{(k)}|I_{s}^{(k)}, M, \Sigma_{s\theta}^{(k)}; H) = \frac{p(I_{s}^{(k)}|s^{(k)}, M, \Sigma_{s\theta}^{(k)}; H)p(s^{(k)})}{\int p(I_{s}^{(k)}|s^{(k)}, M, \Sigma_{s\theta}^{(k)}; H)p(s^{(k)})ds^{(k)}}$$
(10)

(1)

The process is repeated until the parameter estimations satisfy some stopping criterion.

4. IMPLEMENTATION

For a given input image I (Fig. 2a) and a template M (Fig. 2b), a preprocessing step is first applied to obtain a sparse representation. We have used a gradient magnitude representation in our experiments and represent probability distributions by a set of samples with $\mathbf{t}^{(k)}$ being the translation samples (in 2D), $\theta^{(k)}$ the rotation samples and $s^{(k)}$ the scaling layer samples. The probabilities are taken as the weightings, $w_{\theta}^{(k)}$ and $w_{s}^{(k)}$, at the k^{th} iteration of the circuit. The backwards path is run first to obtain imagery hypotheses at each layer. In the three layer case this means running only the backward path of the second and the third layers, which are for rotation and scaling respectively. Therefore the imagery hypotheses for the first layer (for translation in the experiments) are not initialised at this stage of the circuit. Scaling samples $s^{(0)}$ are placed at fixed intervals in range [a, b] and the weights are set as $w_s^{(0)} = 1/s^{(0)}$ (and normalised) to remove the bias towards the centre in the superimposition due to a higher number of pixels concentrated around the centre corresponding to smaller scales. Similarly, rotation samples $\theta^{(0)}$ are initialised at fixed intervals in the range $(0,2\pi]$ and the weights are set to be uniform, summing to 1. Translation samples $\mathbf{t}^{(0)}$ are again initialised at regular intervals in the two dimensional image space. Then, for each k^{th} iteration, inversely scaled samples (illustrated in Fig. 2(c-e)) of the template, M_s , are drawn according to their prior weights $w_s^{(k-1)}$, and the backward observation (mean) is computed as $\overline{M}_s = \sum_j w_{s_j}^{(k-1)} M_s$. This is then fed into the inverse rotation layer and after drawing inversely rotated samples of scaling layer observation, i.e. M_{θ} 's, the backward observation for the rotation layer is computed as $\overline{M}_{\theta} = \sum_{j} w_{\theta_{j}}^{(k-1)} M_{\theta}$. Several observations for different iterations are shown in Fig. 3. For the first



Fig. 2. Input image (a) and the template (b) (template in the experiment is taken form the input image, rotated by 130 degrees and scaled by a factor of 1.2). Several scaling hypothesises (c-e) and rotation hypothesises (f-h).

iteration the sample sets stay the same, so for k = 1, $\mathbf{t}^{(k)} = \mathbf{t}^{(k-1)}$, $\theta^{(k)} = \theta^{(k-1)}$ and $s^{(k)} = s^{(k-1)}$ but for any further iterations an importance sampling scheme is used to obtain a new set of samples as described below:

Resampling: For an iteration number k > 1 and transformation parameter **x** having $N^{(k-1)}$ values in the previous iteration as $\mathbf{x}^{(k-1)} = \{x_1^{(k-1)}, ..., x_{N^{(k-1)}}^{(k-1)}\}$ (for 2D translation layer samples $\mathbf{x}_i^{(k-1)} = (x_{ix}^{(k-1)}, x_{iy}^{(k-1)}), \forall i \le N^{(k-1)})$ with weights $\mathbf{w}^{(k-1)} = \{w_1^{(k-1)}, ..., w_{N^{(k-1)}}^{(k-1)}\}$ where $\sum_{j=1}^{N^{(k-1)}} w_j = 1$ and $N^{(k)}$ new samples to be chosen for the k^{th} iteration are $\mathbf{x}^{(k)} = \{x_1^{(k)}, ..., x_{N^{(k)}}^{(k)}\}$.



Fig. 3. Forward/backward path outputs for a three layer MC-MSC for translation, rotation and scaling (colums) from different iterations (rows) which uses the input image and template given in Fig 2

The Cumulative Function for the $\{k-1\}^{th}$ iteration is constructed as $F(x) = \sum_{\{j:x_j \leq x\}} p_j \ \forall j : j \in \{1, ..., N^{(k)}\}, \ c_j \sim U(0, 1)$ and $i^* = \max\{i : F(x_i) < c_j\}$ then $x_j^{(k)} \sim \mathcal{N}(x_{i^*}^{(k-1)}, \sigma_x^2)$ or $x_j^{(k)} = x_{i^*}^{(k-1)} + \mathcal{N}(0, \sigma_x^2).$

After the new set of samples $\mathbf{t}^{(k)}$, $\theta^{(k)}$ and $s^{(k)}$ is chosen, the forward path is run to compare each of the hypotheses with the templates at each layer and to obtain new weights $w_t^{(k)}$, $w_{\theta}^{(k)}$ and $w_s^{(k)}$. This proceeds as follows: samples of translated input image (patches), I_t , are drawn according to $\mathbf{t}^{(k)}$ and the weights $w_t^{(k)}$ are computed using a likelihood model similar to Eqn.(7). Then the forward observation (mean) in the translation layer is computed as $\overline{I}_t = \sum_j w_{t_j}^{(k)} I_t$ and this is given as the input to the next layer which is for rotation. Samples of rotated input, I_{θ} 's, are then drawn according to $\theta^{(k)}$ with the weights $w_{\theta}^{(k)}$ are computed by Eqn.(7), and the forward observation (mean) in the rotation layer is computed as $\overline{I}_{\theta} = \sum_{j} w_{\theta_{j}}^{(k)} I_{\theta}$. Similar to above, we draw samples of scaled \overline{I}_{θ} $(= I_s$'s) according to $s^{(k)}$ and compute scaling weights, $w_s^{(k)}$ by Eqn.(10). We terminate the search when maximum number of iterations is reached. In the illustrative example given in Figs.2-4, translation layer samples are initially placed uniformly with 5 pixel intervals, rotation samples at 5° intervals in the range of $(0, 2\pi]$ and scaling samples at a step size of 0.01 in the range of [0.8,2]. The forward observations for translation and rotation layer, and the backwards observations for scaling and rotation layers for iterations 1, 5 and 30 are seen in Fig. 3 (again note that the forward scaling and the backward translation layers are not shown as they are not used in the three layer circuit). The probability distributions as represented by the samples are given in Fig. 4. A 2D search space for the translation layer necessitates the probability distribution to be represented with a huge number of samples. As the system refines the weights and starts to converge, the complexity of the marginal likelihoods reduces. Thus, we adopt a data-driven mechanism to set the number of parameters, $N^{(k)}$, in order to reduce computation and sparsify the superpositions. Fig.4 shows the effect of this sample reduction mechanism through the iterations. Another point that should be highlighted in this example is that in the case of multiple



Fig. 4. Updated weights for a three layer MC-MSC of translation, rotation scaling (columns) across different iterations (rows)

hypothesis, Monte Carlo scheme allows peaks to be kept longer allowing recovery in the case, that the system starts to diverge from the true solution owing to collusions, i.e. interference patterns that introduce illusory peaks in the marginal likelihoods. In cluttered scenes, even the gradient magnitude form of the input is not sparse enough for the marginalisation in order to address this. Arathorn uses a large number of orientation channels to reduce this problem [6]. Our strategy is as follows: at earlier iterations ($k = 1 \sim 4$ in the experiments) we first detect the main peaks in the spatial likelihood functions. We then isolate and queue each peak into the three layer MC-MSC. A final comparison yields the best combination of refined parameters. In our experiments, for the case of detecting 9 objects in a cluttered scene (Fig.5), 15 different combinations of rotations and scalings are applied to the templates making up a total number of 135 different cases. Parameters were chosen randomly in a range of [0, 360) degrees for rotation and a range of [0.8, 1.7] for scaling. Our system achieved 79% for the correct localisation whereas standard MSC (on gradient images) suffered severely from collusions and achieved only 21%. For MC-MSC mechanism absolute mean errors for the rotation were 2.02 degrees for rotation and 0.06. For the correct detections, the final similarity of transformed templates to the identified locations in the input scene is computed by a normalised cross correlation which ranged from 0.85 to 0.95. If the same number of samples had been employed on a discrete grid, the best precision of the MSC would be constrained to the grid locations at 5 pixels translation, 5° rotation and 0.05 scale increments.



Fig. 5. Input image for 'queuing' in MC-MSC experiments

5. CONCLUSIONS

Pattern recognition through a series of marginalisations is an attractive idea as real imagery often requires a high dimensional space to be explored. Although a practical implementation has not been proposed until recently [1], the concept of Bayesian marginalisation for image transformations, which allows the incorporation of parallel search (a behaviour observed in the human visual system) to many computer vision problems. Using a Monte Carlo sampling approach in such a system increases the adaptability and efficiency leading to more accurate approximation of transformation values by a smaller number of computations.

Marginalisation in the form of superpositions aids the search for multiple variables by reducing the dimensionality of the search space, for the problems otherwise requiring a fully sequential search. However as the complexity of information included in one marginalisation increases, the MSC can fail to converge to the global maximum. Motivated by biological vision, an ideal trade-off is obtained when this parallel search mechanism is supported with a few sequential steps increasing the complexity by a small amount. This is referred as 'queuing' in this paper. Queuing incorporates all the advantages of the marginalisation and increases the efficiency by partitioning the information in the marginalisations guided directly by the system. Future work includes studying the conditions under which marginalisation through image superimposition works, i.e. where the sparsity assumptions break down.

6. REFERENCES

- David W. Arathorn, Map-seeking circuits in visual cognition a computational mechanism for biological and machine vision, Stanford University Press, 2002.
- [2] Tomáš Gedeon and David W. Arathorn, "Convergence of map seeking circuits," J. Math. Imaging Vis., vol. 29, no. 2-3, pp. 235–248, 2007.
- [3] Jan Jelinek, "Analysis of the map-seeking circuit," in Intelligent Computing: Theory and Applications V. Edited by Priddy, Kevin L.; Ertin, Emre. Proceedings of the SPIE, Volume 6560, pp. 656008, May 2007, vol. 6560.
- [4] Thomas L. Thornton and David L. Gilden, "Parallel and serial processes in visual search," *Image Processing*, 2007. *ICIP* 2007. *IEEE International Conference on*, vol. 114, no. 1, pp. 71–103, Jan. 2007.
- [5] Jun S. Liu and Rong Chen, "Sequential monte carlo methods for dynamic systems," *Journal of the American Statistical Association*, vol. 93, pp. 1032–1044, 1998.
- [6] David W. Arathorn, Personal Communication.