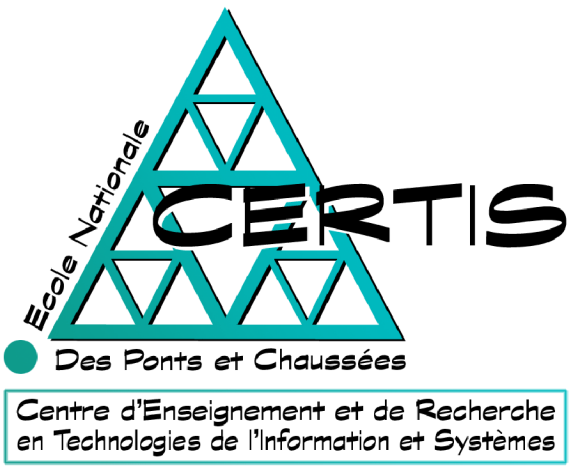
Combinatorial optimization for electrode labeling of EEG Caps

Mickaël Péchaud
Renaud Keriven
Théo Papadopoulo
Jean-Michel Badier

Research Report 07-36
May 2007



CERTIS, ENPC,
77455 Marne la Vallee, France,
http://www.enpc.fr/certis/

Combinatorial optimization for electrode labeling of EEG Caps

Étiquetage d'électrodes de casque EEG par optimisation combinatoire

Mickaël Péchaud[1]
Renaud Keriven
Théo Papadopoulo
Jean-Michel Badier

[1]Odyssée Lab, École Normale Supérieure, École des Ponts, INRIA
45, rue d'Ulm - 75005 Paris - FRANCE

[4]CERTIS, ENPC, 77455 Marne la Vallee, France, http://www.enpc.fr/certis/

Abstract

An important issue in electroencephalographiy (EEG) experiments is to measure accurately the three dimensional (3D) positions of the electrodes. We propose a system where these positions are automatically estimated from several images using computer vision techniques. Yet, only a set of undifferentiated points are recovered this way and remains the problem of labeling them, i.e. of finding which electrode corresponds to each point. We propose a fast and robust solution to this latest problem based on combinatorial optimization. We design a specific energy that we minimize with a modified version of the Loopy Belief Propagation algorithm. Experiments on real data show that, with our method, a manual labeling of two or three electrodes only is sufficient to get the complete labeling of a 64 electrodes cap in less than 10 seconds.

Résumé

Un des problèmes importants lors d'une expérience d'électroencéphalographie (EEF) est de mesurer avec précision la position tridimensionnelle des éléctrodes du casque. Nous proposons un système permettant d'estimer ces positions automatiquement à part d'un certain nombre de prises de vues, utilisant des méthodes standarts de vision par ordinateur. Cependant, ces techniques permettent seulement de retrouver un ensemble de points indifférenciés, et il reste à étiqueter ces points, c'est à dire à retrouver à quelle électrode correspond chaque point.

Nous proposons une méthode rapide et robuste, basée sur des méthodes d'optimisation combinatoire pour résoudre ce problème. Le problème est formulé sous forme énergétique, et l'opimisation est effectuée à l'aide d'une version modifiée de l'algorithme Belief Propagation.

Des expériences sur des jeux de données réelles montrent que notre méthode permet de retrouver un étiquetage complet en moins de 10 secondes pour un casque à 64 électrodes, en spécifiant manuellement 2 ou 3 électrodes.

Contents

1 Introduction

Electroencephalography (EEG) is a widely used method for both clinical and research purposes. Clinically, it is used e.g. to monitor and locate epilepsy, or to characterize neurological disorders such as sleeping or eating disorders and troubles related to multiple sclerosis. Its main advantages are its price compared to magnetoencephalography (MEG), and its very good time resolution compared e.g. to fMRI. Conventionally, EEG readings were directly used to investigate brain activity from the evolution of the topographies on the scalp. Nowadays, it is also possible to reconstruct the brain sources that gave rise to such measurements, solving a so-called inverse problem. To this purpose, it is necessary to find the electrode positions and to relate them to the head geometry recovered from an anatomic MRI. Current techniques to do so are slow, tedious, error prone (they require to acquire each of the electrodes in a given order with a device providing 3D coordinates[17]) and/or quite expensive (a specialized system of cameras is used to track and label the electrodes[24]). Our goal is to provide a cheap and easy system for electrode localization based on computer vision techniques.

In modern EEG systems, the electrodes (64, 128 or even 256) are organized on a cap that is placed on the head. To each cap system, it is possible to associate a "model" describing the electrodes, their labels and their relative positions. One way to obtain such a model is e.g. to use one of the standard techniques once to obtain a mesh of labeled electrodes. This model is then used as a template to drive a labeling procedure that starts with some roots provided by the user and then uses the similarity between the neighboring properties between the measured electrodes and those provided by the model. Our system takes as inputs multiple pictures of the head wearing the cap from various positions. As a preliminary step, electrodes are localized and their 3D positions are computed from the images by self-calibration (a technique that recovers the cameras' positions from the image information [8]) and triangulation. These are standard techniques that can provide 3D point coordinates with a quite good accuracy. Remains the problem of electrode identification which labels each 3D position with the name of the corresponding electrode. Finding a solution to this last problem is the focus of this paper. Note, that a good labeling software can also improve current systems by removing acquisition constraints (such as the recording of the electrodes in a given order) and by providing better user interfaces.

We propose a method that recovers this labeling from just a few (two or three) manually annotated electrodes. The only prior is a reference, subject independent, 3D model of the cap. Our framework is based on combinatorial optimization (namely on an extension of the Loopy Belief Propagation algorithm[21]) and is robust to soft deformations of the cap caused both by sliding effects and by the variability in subjects' head geometry.

2 Problem definition

The inputs of our method consist of:

- a template EEG cap model providing labeled electrodes, along with their 3D positions (in fact, as we will explain further, an important feature of our method is that only the distances between close electrodes are used). $\mathcal{L}$ will denote the set of labels (e.g. $\mathcal{L} = \{Fpz, Oz, \cdots\}$), and $C = \{C_l,\ l \in \mathcal{L}\}$ will be their corresponding 3D positions. C_l could be for example the average position of electrode l among a variety of prior measures. However, in our experiments, it was just estimated on one reference acquisition.

- the measured 3D positions of the electrodes to label, obtained by 3D reconstruction from images. We will denote by $M = \{M_i,\ i \in [1..n]\}$, these n 3D points.

The output will be a labeling of the electrodes, i.e. a mapping ϕ from $[1..n]$ to $\mathcal{L}$. Note that n could be less than the total number $|\mathcal{L}|$ of electrodes in cases where some electrodes are of the cap are not used.

3 Motivation

In this section, we discuss other possible approaches for the electrode labeling problem. As it will be detailed in section 6, we have tried some of these methods without any success. This will motivate our energy-based combinatorial approach. A simple method could consist on a 3D registration step, followed by a nearest-neighbor labeling. Let T be a transformation that sends M into the spatial referential of C. A straight labeling could be:

$$\phi(i) = \arg\min_{l \in \mathcal{L}} d(C_l, T(M_i))$$

where $d(A, B)$ denotes the Euclidean distance between points A and B. Actually, we first tested two direct ways of obtaining an affine transformation T:

- *moment-based affine registration*: in this case, we computed first and second order moments of the sets of points M and C and choose T as an affine transformation which superimposes these moments.

- *4 points manual registration*: here, we manually labeled 4 particular electrodes in M and took for T the affine transformation which exactly sends these 4 electrodes to the corresponding positions in C.

As explained in section 6, we observed that these two approaches give very bad average results. One could argue that this might be caused by the quality of the registration. A solution could be to use more optimal affine registration methods,

like Iterative Closest Points[28, 3]. Yet, a close look at what caused bad labeling in our experiments, reveals that this would not improve the results. The main reasons are indeed that (i) the subject whose EEG has to be labeled does not have the same head measurements than the template, and moreover that (ii) the cap is a soft structure that shifts and twists from one experiment to another.

It is clear that only a non-rigid registration could send M close to C. However, modeling the problem in term of space deformation is not suitable. For instance, a Thin-Plate Spline[5, 12] based algorithm would not be adapted. Actually, a more suitable framework could be a deformable shape matching one. We could see our problem as a shape registration one, based on shape deformation and *intrinsic* shape properties[25], rather than on deforming the ambient space in order to make the shapes match. Because of the topology of the electrodes on the cap, relations between points are also of importance. In that sense, our problem is close to the one investigated by Coughlan et al. [7, 1], which they solve recovering both deformations and soft correspondences between two shapes. Yet, in our case, we see two main differences: (i) labeling, rather than shape matching, is the key issue, and (ii) enforcing relational constraints between points are more important than regularizing deformations. For these reasons, we propose a method based on optimal labeling for which the only (soft) constraints are the distances between nearby points, without modeling any deformation.

Moreover, we optimize our objective function thanks to a accelerated version of the original Loopy Belief Propagation algorithm[21].

In the remaining of the article, we first state our model and the associated energy; we then discuss our choice for an energy minimization algorithm. Finally, we validate our method giving qualitative and quantitative results on real experiments.

4 Proposed framework

The complete pipeline of our system is depicted figure 1. As we already explained, we do not consider here the 3D reconstruction step, but only the labeling one. From the measured data M, we construct an undirected graph $G = (V, E)$, where $V = [1..n]$ is the set of vertices and E a certain set of edges which codes the relations between nearby electrodes. As it will become clear in the following, the choice of E will tune the rigidity of the set of points M. Practically, the symmetric k-nearest neighbors or all the neighbors closer than a certain typical distance, are two valid choices. Given an edge $e = (i, j) \in E$ for $i \in V$ and $j \in V$, we denote by $d_{ij} = d(M_i, M_j)$ the distance between points M_i and M_j in the measured data and by $\tilde{d}_{ij} = d(C_{\phi(i)}, C_{\phi(j)})$ the reference distance between the electrodes $\phi(i)$ and $\phi(j)$. In order to preserve in a soft way the local structure of the cap, we

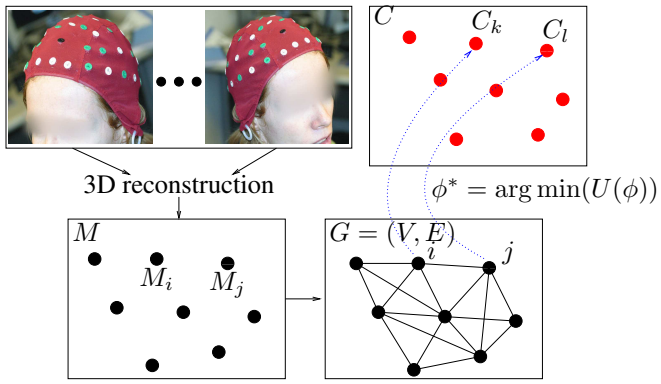


Figure 1: Complete pipeline : we obtain 3D positions M (bottom left) by reconstruction from several (usually 6) pictures (top left). A graph G then is constructed from these positions (bottom right). Considering a template cap and associated positions C (top right), we label the measured electrodes by estimating $\phi^* = \arg\min(U(\phi))$. In this example, $\phi(i) = k$, $\phi(j) = l$.

propose to simply minimize the following energy:

$$U(\phi) = \sum_{(i,j)\in E} \rho(d_{ij}, \tilde{d}_{ij}) \tag{1}$$

where ρ is a cost-function which penalizes differences between the observed and template distances. Note that, whereas the global one-to-one character of ϕ is not explicitly enforced by this model, the local rigidity-like constraints enforce it. Graph rigidity theory is a very complex domain (see for example [4] as an introduction), beyond the purpose of this article.

Following the classical framework of Markov Random Fields (MRF) [18, 2, 10], this can be rewritten as maximizing the following function:

$$P(\phi) = exp(-U(\phi)) = \prod_{(i,j)\in E} exp(-\rho(d_{ij}, \tilde{d}_{ij})) = \prod_{(i,j)\in E} \Psi_{i,j}(\phi(i), \phi(j)) \tag{2}$$

Normalizing P by dividing by the sum over all the possible mappings ϕ, yields a *Gibbs distribution* over a MRF derived from graph G with $\mathcal{L}$ as the set of possible labels each vertex. The problem is thus reduced to the classical case of finding a *Maximum A Posteriori* (MAP) configuration of a Gibbs distribution:

$$p(\phi) = \frac{1}{K} \prod_{i\in V} \psi_i(\phi(i)) \prod_{(i,j)\in E} \psi_{i,j}(\phi(i), \phi(j)) \tag{3}$$

where K is a normalizing constant and $\psi_i(\phi(i)) = 1$ in our case.

5 Energy minimization

The problem of finding a MAP configuration of a Gibbs distribution being NP-complete [15], we cannot expect to get an algorithm that optimally solves every instance of the problem. Since the seminal work of Geman & Geman [10], who proposed an algorithm that warrants the probabilistic convergence toward the optimal solution – however with an unreasonable run-time – several methods have been investigated to maximize general distributions like (3). Among these, minimal-cut based methods (often referred to as *GraphCuts*), introduced in computer vision and image processing by [11], has received many attention (see [14, 6]). These methods can achieve global optimization for a restricted class of energies[13]. For more general energies, approximations were proposed [23]. As we experimented[19], these approximations fail to recover a correct labeling in our problem, which belongs to a class of multilabel problems that are not easily tackled by *GraphCuts*.

As a consequence, we opted for a completely different but widely spread algorithm, namely *Belief Propagation* (BP), and more precisely for its variant adapted to graphs: *Loopy Belief Propagation* (LBP). Please see [9] for a recent reference. Briefly, it consists in propagating information through the edges of the graph: each node i sends *messages* to its neighbors k, measuring the estimated label of k from its own point of view. Messages are passed between nodes iteratively until a convergence criterion is satisfied. This algorithm is neither guaranteed to converge nor to converge to an optimal solution. However, it behaves well in a large variety of early vision problems. Empirical and theoretical convergence of this family of methods were studied for instance in [20, 27].

Actually, we designed for this work an original and faster version of LBP. It is an improved version of LBP based on the idea of [16].

Let us first explain classical LBP algorithm.

5.1 LBP

Loopy Belief Propagation (LBP) algorithm [9] is a widely used method to find approximate solutions to the MAP problem when the sub-modularity condition is not fulfilled. It consists in propagating information through the vertices of the mesh seen as a graph: roughly speaking, each node i sends *messages* to its neighbors k, measuring the estimated label of k from the point of view of i.

The LBP algorithm is derived from an exact algorithm working on trees called *Belief Propagation* (BP) or *Max-Product* algorithm [22]. In the original BP, messages measuring belief in a local labeling propagate from the leafs to the root of the tree. Then a backward pass is computed in which label that maximizes the belief is chosen at each node, knowing the label of the father.

Let us introduce some notations :

r the root of the tree, s the application that maps a node to its sons and f the application that maps a node to its father. L is the set of the leafs of the tree.

$m_{i\to j}$ will denote the message passed by node V_i to node V_j. $m_{i\to j}(l)$ is a measure of how confident node V_i is that node V_j is given the l label, i.e. $\phi(V_j) = l$.

We denote by $b_i(l_i, l_{f(i)}) = \psi_i(l_i)\psi_{f(i)i}(l_{f(i)}, l_i)\prod_{j\in s(i)} m_{j\to i}(l_i)$ for $l_i \in C$ and $l_{f(i)} \in C$ the joint belief that node $V_{f(i)}$ is assigned label $l_{f(i)}$ and node V_i is assigned label l_i.

The BP algorithm is described in algorithm 1.

Algorithm 1 Belief Propagation

$K = L$
FORWARD PASS
while $V_i \in K$ **do**
  remove V_i from K
  compute $m_{i\to f(i)}(l_{f(i)}) = max_{l_i}(b_i(l_i, l_{f(i)}))$ for all $l_{f(i)} \in C$.
  compute $\delta(l_{f(i)}) = argmax_{l_i}(b_i(l_i, l_{f(i)}))$ for all $l_{f(i)} \in C$.
  if all sons of $V_{f(i)}$ have been treated, add $V_{f(i)}$ to K
end while
BACKWARD PASS
$\bar{\phi}_r = \delta_r$
$K = s(r)$
while $V_i \in K$ **do**
  remove V_i from K
  compute $\bar{\phi}_i = \delta_i(\bar{\phi}_{f(i)})$
  $K = K \cup s(V_i)$
end while
return $\bar{\phi}$

When the graph is not a tree, the ordered treatment of BP is impossible to apply. However, disregarding the relation of paternity of the nodes, it is still possible to pass messages from nodes to nodes in the graph. A belief can also be computed the same way as for BP. The idea of LBP is then to apply the message passing simultaneously or sequentially to all the neighboring nodes of the graph. A stopping criterion is then to be defined - usually a convergence criterion or a fixed number of iterations.

Let us adapt slightly the notations and denote by $N(i)$ the set of neighbor nodes of V_i.

$m^t_{i\to j}$ is the message passed by node V_i to node V_j at time t. Let $b^t_i(l_i, l_j) =$

$\psi_i(l_i)\psi_{ij}(l_i,l_j)\prod_{k\in N(i),k\neq j} m^t_{k\to i}(l_i)$ for $(l_i,l_j)\in C^2$ be the joint belief for neighbor nodes V_i and V_j. Finally, let $b^t_i(l_i)=\psi_i(l_i)\prod_{k\in N(i)} m^t_{k\to i}(l_i)$ be the $belief\ vector$ at node V_i and time t (taking into account all the neighbors of node V_i).

This leads to algorithm 2.

Algorithm 2 Loopy Belief Propagation

set $m^0_{p\to q}(l_q)=1$ for all $(p,q)\in E$.
for $t=1, t\leq T, t++$ **do**
  for all (i,j) in V **do**
    $m^t_{i\to j}(l_j)=max_{l_i\in C}(b^{t-1}_i(l_i,l_j))$
  end for
end for
return $\bar{\phi}_i=argmax_{l_i\in C} b^T_i(l_i)$

This algorithm is neither guaranteed to converge nor to converge to an optimal solution. However, it behaves well in a large variety of early vision problems. Empirical and theoretical convergence of this kind of methods were studied for instance in [20] and [27].

Notice that the complexity of one step of this algorithm is basically $|(C)|^2|E|$ where $|E|$ is the number of edges of the graph.

5.2 Improving belief propagation

Several methods have been proposed to improve both the convergence and the quality of results obtained by LBP algorithm. [26] proposed a slightly different algorithm based on a different theoretical framework with interesting convergence properties. More recently, [16] proposed an interesting modification of LBP based on label pruning according to current belief at each node, and on a choice of a priority order for covering all nodes. But, their method show a greedy behavior, since a label cannot appear again once it has been pruned.

A new intermediate and simpler version of LBP based on label pruning is proposed here. It is based on the idea that if a label is very unlikely for a given vertex, it ought to be useless to use this label for the calculation of the outgoing messages for this vertex. Hence, after each step, the *belief vector* $b^t_i(l_i)$ is computed for each node as well as its maximum and minimum values M^t_i and m^t_i. Then, each label with a belief lower than the geometric mean g^t_i of m^t_i and M^t_i is declared *inactive* for the next iteration only, e.g. it won't be considered as a candidate label in computing outgoing messages toward the neighbors of V_i (notice that the choice of the mean is somewhat arbitrary. It should be adapted to the structure of the belief

vector. For our application, we didn't notice effect of the choice of a threshold between 0.5 and 0.8 over speed nor quality of results).

Let us denote by Act_i^t the set of active labels of V_i computed at iteration t. Our method is described algorithm 3.

Algorithm 3 Fast Loopy Belief Propagation

set $Act_i^0 = C$ for all $V_i \in V$
set $m_{p\to q}^0(l_q) = 1$ for all $(p,q) \in E$.
for $t = 1, t \leq T, t++$ **do**
  for all (i,j) in V **do**
    $m_{i\to j}^t(l_j) = max_{l_i \in Act_i^{t-1}}(b_i^{t-1}(l_i, l_j))$
    set $Act_i^t = \{l_i : l_i \geq g_i^t\}$
  end for
end for
return $\bar{\phi}_i = argmax_{l_i \in C} b_i^T(l_i)$

The $|C|^2$ factor for each edge in the complexity for one step is then replaced by a $|C||C'|$ where $|C'|$ is the number of *active* labels of the original node.

6 Experiments

We used 6 sets of 63 electrodes. Each set consists of 63 estimated three dimensional points, acquired on different subjects with the same EEG cap and manually labeled. To test our algorithm as extensively as possible, we ran the algorithm on each set, taking successively each of the other sets as a reference. We hence simulated 30 different pairs (M, C). At least one electrode in M was manually labeled (see further).

E was chosen the following way : we first estimated a typical neighbor distance by computing the maximum of the nearest neighbor distance for all electrodes in M, and then considered as belonging to E, every pair of distinct electrodes within less than three times this distance. In order to accelerate and enforce convergence, we used the three following technical tricks:

- we used our modified LBP algorithm[19]
- we added a classical momentum term ([20])
- denoting by V_f the subset of V of the manually labeled electrodes, we added the set of edges $V_f \times V$ to E, allowing accurate information to propagate quickly in the graph.

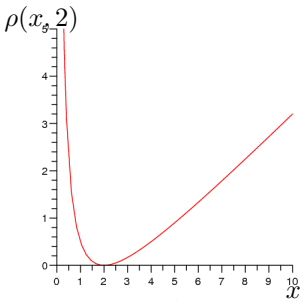


Figure 2: $\frac{x}{y+\epsilon} + \frac{y}{x+\epsilon} - 2$ for $y = 2$, $\epsilon = 1e^{-4}$

Although non indispensable, this led to a smaller number of non converging optimization.

On our dataset, our modified LBP algorithm led to a mean run-time improvement by a factor over 2, while neither the convergence nor the quality of the results were affected compared to classical LBP. The mean run-time was less than 11s on a standard 3GHz PC using our modified LBP algorithm.

Prior knowledge of the label of some (at least one) electrodes was introduced. Such prior information on the solution fits very naturally with MRF framework and could be easily extended to more sophisticated constraints (many nodes known, nodes known to be in a subset of the labels, ...).

The cost-function ρ was of the form $\rho(x, y) = \frac{x}{y+\epsilon} + \frac{y}{x+\epsilon}$ where ϵ is a small positive constant. We did not notice sensitivity with respect to this choice, as far as the following key conditions are fulfilled: (i) penalizing differences between x and y and (ii) penalizing small values of x or y. This latest condition enforces (yet does not warrant) a one-to-one mapping ϕ. (fig. 2).

Different experiments where carried out. First, the prior consisted in manually labeling electrodes Fpz, Oz, and $T8$. In that case, our method recovers all the electrodes, which was, as expected, not at all the case with an affine registration+nearest neighbor approach (see figure 3). Actually, we observed that labeling $(Oz, T8)$ seems sufficient. Yet, without any further data, we do not consider that labeling two electrodes only is reliable. Figure 5 shows a result on a case where affine registration does not work and the final 3D reconstruction with our method.

To demonstrate the robustness of our algorithm, we also tested hundreds of other conditions, in which 1, 2 or 3 randomly chosen electrodes were "manually" labeled. Non-convergence was only observed for non reasonable choices of "manually" labeled electrodes: indeed, if they are chosen on the sagittal medium line, there is an undetermination due to the left-right symmetry of the cap. This does

not occur when the electrodes are set by a human operator. The classification error rates are low (see figure 3 again) but not negligible. This makes us plead for a manual labeling of two or three fixed and easy to identify electrodes, e.g. $(Fpz, Oz, T8)$.

Finally, we also successfully tested cases for which $n < |\mathcal{L}|$, i.e. when some electrodes are missing : if a few electrodes were forgotten in the 3D reconstruction process, our algorithm should still be able to label the detected ones. This should allow us to find which electrodes were forgotten, to compute their approximate 3D position from the template cap model and to use this information to detect them back in the pictures. To carry our experiments, we removed randomly from 1 to 10 electrodes in the data sets to be labeled. Labelisation was performed using the $(Fpz, Oz, T8)$ prior as explained above. Results are synthetized figure 4. .

	NC	misclassified labels
Affine registration (moment based)	-	48.7%
Affine registration (4 manual points)	-	21.3%
Our method - $(Fpz, Oz, T8)$ manually labeled	0%	0%
Our method - $(Oz, T8)$ manually labeled	0%	0%
Our method - 3 random electrodes labeled	0%	0.03%
Our method - 2 random electrodes labeled	0.3%	0.2%
Our method - 1 random electrode labeled	4.2%	3,7%

Figure 3: Classification errors. NC gives the percentage of instances of the problem for which our method did not converge. Misclassified labels percentages are estimated only when convergence occurs. Misclassified labels include very rare cases were ϕ wasn't one-to-one.

7 Conclusion

Experiments show that our framework leads to fast, accurate and robust labeling on a variety of data sets. We consider providing on the WEB in a near future an complete pipeline including our algorithm - ranging from 3D reconstruction of electrodes to their labeling. Such a system would only require a standard digital camera and would imply minimal user interaction (manually labeling three electrodes).

Note that the flexibility of our MRF formulation allows different priors. We plan for instance to use the color of electrodes on the images as a further prior for labeling. This could lead to a fully automated system, where no user interaction would be required.

missing electrodes	mislabeled electrodes
1	0%
2	0%
3	0.01%
4	0.02%
5	0.02%
6	0.04%
7	0.04%
8	0.3%
9	1.1%
10	1.1%

Figure 4: Results with missing electrodes.

References

[1] A.Rangarajan, J.M. Coughlan, and A.L. Yuille. A bayesian network framework for relational shape matching. In *9th IEEE ICCV*, pages 671–678, 2003.

[2] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal Royal Statis. Soc.*, B-148:192–236, 1974.

[3] P.J. Besl and N.D. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, 1992.

[4] B.Hendrickson. Conditions for unique graph realizations. *SIAM J. Comput.*, 21(1):65–84, 1992.

[5] F.L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Trans. PAMI*, 11(6):567–585, 1989.

[6] Y. Boykov, O. Veksler, and R. Zabih. Markov random fields with efficient approximations. In *CVPR '98*, page 648, Washington, DC, USA, 1998. IEEE.

[7] J.M. Coughlan and S.J. Ferreira. Finding deformable shapes using loopy belief propagation. In *7th ECCV*, pages 453–468, 2002.

[8] O. Faugeras, Q.T. Luong, and T. Papadopoulo. *The Geometry of Multiple Images*. MIT Press, 2001.

[9] P. Felzenszwalb and D. Huttenlocher. Efficient belief propagation for early vision, 2004.

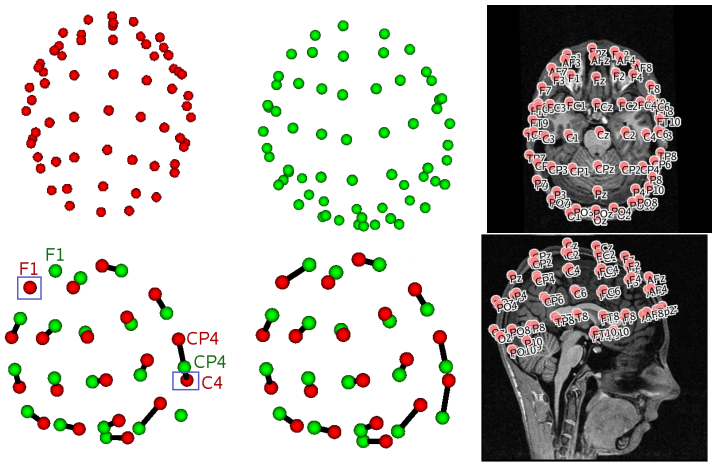


Figure 5: A sample result. M is in red and C in green. Top left: 63 estimated 3D electrodes positions. Top center: reference. Bottom left: subset of a labeling with the moment based algorithm; C4 is wrongly labeled CP4, and F1 is labeled F3 (not shown). Bottom center: a subset of correct correspondences retrieved by our algorithm. Top and bottom right: full labeling retrieved by our algorithm, superimposed with anatomical MRI

[10] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Trans. PAMI*, 6(6):721–741, Nov. 1984.

[11] D.M. Greig, B.T. Porteous, and A.H. Seheult. Exact maximum a posteriori estimation for binary images. *J. R. Statist. Soc. B*, 51:271–279, 1989.

[12] H.Chui and A. Rangarajan. A new algorithm for non-rigid point matching. In *CVPR*, pages 2044–2051, 2000.

[13] H. Ishikawa. Exact optimization for markov random fields with convex priors, 2003.

[14] H. Ishikawa and D. Geiger. Mapping image restoration to a graph problem, 1999.

[15] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? In *ECCV (3)*, pages 65–81, 2002.

[16] N. Komodakis and G.Tziritas. Image completion using global optimization. In *CVPR '06*, Washington, DC, USA, 2006. IEEE Computer Society.

[17] D. Kozinska and K. Nowinski. Automatic alignment of scalp electrode positions with head mrivolume and its evaluation. In *Engineering in Medicine and Biology, BMES/EMBS Conference*, Atlanta, Oct 1999.

[18] P. Lévy. Chaînes doubles de markov et fonctions aléatoires de deux variables. *C.R.Académie des sciences*, 226:53–55, 1948.

[19] M.Pechaud, R.Keriven, and T.Papadopoulo. Combinatorial optimization for electrode labeling of EEG caps. Technical Report 07-32, CERTIS, July 2007.

[20] K.P. Murphy, Y. Weiss, and M.I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Fifteenth Conference on Uncertainty in Artificial Intelligence*, pages 467–475, 1999.

[21] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, Inc., 1988.

[22] J. Pearl. *Probabilistic Reasonning in Intellignet System:Networks of Plausible Inference*. Morgan Kaufmann Publishers Inc., 1988.

[23] A. Raj and R. Zabih. A graph cut algorithm for generalized image deconvolution. In *ICCV '05*, pages 1048–1054, Washington, DC, USA, 2005. IEEE.

[24] G. S. Russell, K. J. Eriksen, P. Poolman, P. Luu, and D. M. Tucker. Geodesic photogrammetry for localizing sensor positions in dense-array eeg. *Clinical Neurophysiology*, 116:1130–1140, 2005 (see also http://www.egi.com/c_gps.html).

[25] T.B. Sebastian, P.N. Klein, and B.B. Kimia. Alignment-based recognition of shape outlines. In *4th International Workshop on Visual Form*, pages 606–618, 2001.

[26] V.Kolmogorov. Convergent tree-reweighted message passing for energy minimization. Technical Report MSR-TR-2005-38, 2005.

[27] Y. Weiss and D. Freeman. On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs. *IEEETIT*, 47, 2001.

[28] Z. Zhang. iterative point matching for registration of free-form curves. Technical Report RR-1658, INRIA, 1992.