Building and Tracking Hierarchical Geographical & Temporal Partitions for Image Collection Management on Mobile Devices

Antoine Pigeau, Marc Gelgon

Building and Tracking Hierarchical Geographical & Temporal Partitions for Image Collection Management on Mobile Devices

A. Pigeau and M.Gelgon
LINA FRE 2729 CNRS / ATLAS group, INRIA
2, rue de la Houssinière
BP 92208
44322 Nantes cedex 03 - France
email: *firstname.lastname*@univ-nantes.fr

Abstract

Usage of mobile devices (phones, digital cameras) raises the need for organizing large personal image collections. In accordance with studies on user needs, we propose a statistical criterion and an associated optimization technique, relying on geo-temporal image metadata, for building and tracking a hierarchical structure on the image collection. In a mixture model framework, particularities of the application and typical data sets are taken into account in the design of the scheme (incrementality, ability to cope with non-Gaussian data, with both small and large samples). Results are reported on real data sets.

1 Introduction

Through the daily use of mobile devices (phones, digital cameras), large personal image collections are currently being built by consumers. As it is essential to provide these users with solutions for retrieving pictures efficiently among usually several thousands, the corresponding research sub-field of multimedia indexing is currently attracting much interest. The Nokia Lifeblog product [16] and Microsoft MyLifeBits research prototype [11] are two recent answers from industry. Particularities of the task, compared to more classical research on multimedia content-based retrieval, largely come from the image meta-data provided by the acquisition device (time, geolocation, camera settings) and the querying/browsing criteria preferred by users. Studies on this latter point, reported in [6, 20, 25, 26], conclude, as one would expect, that social interaction/events, time and places are the appropriate cues to trigger memories.

In this field of consumer images, some work has addressed image content-based supervised classification (e.g. distinction between indoor and outdoor [14], face-based characterisation [9] which is now a reasonnable task to implement on PDAs with recent low-cost algorithms [24]). In contrast, we focus, in this paper, on the sole use of temporal and geolocation meta-data attached to each picture. We assume location are coordinates provided by a GPS/E-OTD type of equipment, i.e. the data is a stream of $\{(t,(x,y)) \in \mathbb{R} \times \mathbb{R}^2\}$ elements. Throughout, although we mention the example of an image collection, since this is the major current applicative need, the proposal is not technically tied to a particular type of document. The contribution put forward is a technique (criteria and algorithm) for automatically building a hierarchical organization of images, based on their time and geolocation stamps. Such a structure obviously assumes that the generative process of pictures frequently exhibits such geo-temporal clusters and often in a hierarchical fashion. In other words, the task may be viewed as the "inverse problem" of recovering of meaningful episodes of the user's life, given images provided during these episodes. The hierarchies of partitions are built incrementally, as data flows in, since the image acquisition and collection browsing phases are highly interleaved. The goal of extracting such distinct spatial and temporal structures is supported by the following reasons :

1. the spatial and temporal axes employed for structuring are clear and familiar to users (the diary and map metaphores) compared to e.g. browsing according to image color features. Still, no map browsing is intended here, time-oriented views can be built on a location-based structure;

2. at least one of the two tags of the sought document is often remembered by the user [25], but he may approach his goal by iteratively switching between the temporal and spatial views, depending on how viewing intermediate images during the search trigger re-orientation;

3. browsing, rather than querying, enables the user to navigate into a personal multimedia diary/photo album without having a particular target-picture in mind (as a passtime or to get an overview);

4. browsing along these axes is feasible from a mobile phone which, despite its limited man-machine interaction (input and display), has better availability than the desktop PC;

5. the structure obtained can serve system-level efficiency (speed, device consumption). In our context, an implicit goal is to minimize the number of "useless" pictures that are displayed when browsing, since fast successive display of many color images through naïve timeline browsing is rather power consuming. Overall, the proposed structure also helps prefetch and cache images in an effective way, as we proposed in [15].

The remainder of this section outlines the principles and characteristics of the proposed contribution. We formulate the recovery of the image collection hierarchical spatial and temporal structures as an unsupervised classification task.

Distinct hierarchical classifications are built for time and space, respectively from temporal and geolocation data, but are tackled with almost identical techniques. We opt for the mixture model framework, in which probabilistic models are associated both to class-conditional probability densities and to data-to-class assignments. This choice grants two advantages :

- it breaks the combinatorial explosion inherent to data grouping problems;
- it suits well the incremental nature of the task, since data-to-class assignments may evolve in a flexible way as new data streams in, using a light predict/update mechanism.

More precisely, the scheme has the following features :

1. determination of meaningful spatial and temporal groups relies on probabilistic modelling and a statistical optimality criterion that exhibit several good properties with respect to the task : structure complexity trade-off, robustness to non-Gaussianity of clusters, coping with small samples);
2. classifications are built in an incremental manner (i.e., on-line with regarding to arrival of data) using a search procedure which adds notably split and merges to the EM algorithm, thus avoiding some poor local optima of the abovementioned criterion and enabling the update structural evolution (including the number of clusters);
3. a hierarchy of clusterings is built bottom-up and updated over time at low cost;
4. in all of these phases, critical arbitrary parametrization is largely avoided.

Our main contributions in this paper are the proposed procedures of the points 2 and 3, leading to an incremental and hierarchical algorithm.

A necessary side-task in the real application is the determination of a small subset of "visually representative" images from the images contained in a group. We do not address this herein, as there exist effective techniques proposed in the context of video summarization (eg. [23]), that could be employed.

The remainder of this paper is organized as follows. Section 2 surveys work related to temporal and/or geolocation-based structuring for the application at hand. Section 3 then discloses the technical proposal : the probabilistic modelling and associated optimization phases on the finest-scale layer of the hierarchy are described. The process for building and tracking a hierarchy is then presented. Section 4 provides experimental results. Finally, the work is summarized and perspectives are sketched in section 5.

2 Related work

Time and geolocation annotations on personal images have been introduced in many services on the market, for the time being ignoring automated organization (Microsoft World Wide Media eXchange system [21], Picasa "Hello"

(www.hello.com), Cognima (www.cognima.com)). Whether image collections should be manually or automatically organized (despite possible errors) is still under debate [16], but we advocate, with many others quoted below in this section, that automation is more beneficial than harmful.

For existing automated schemes at research stage, time stamp has long been a favourite since it is an intuitively appealing, cheap and reliable measurement. Segmenting the sequence of time stamps has been viewed in [12, 13, 19] as the incremental detection of gaps. Some thresholding sets the definition of a "meaningful" gap. Time structuring can also be combined with image features [5, 13], or the camera settings [8]. Personal diary structuring based on location was proposed in [10], but measuring location continuously in time (rather than based on punctual picture acquisition). Partitions are extracted at multiple scales, based on a piece-wise parametric trajectory model. By this means, one attempts to recover significant temporal episodes and areas. A work close in spirit is [2], although the modelling formalism differs. The recent work described in [1] proposes some elementary automated organization, but contributes mainly in the browsing mechanisms.

The work closest to the present paper is [17], which also organises an image collection hierarchically, based on time and location clusters. To our understanding, their work incorporates a series of rules derived from user expectations. Although these rules are very appropriate (the study of joint time/space criteria is particularly appealing), they seem to imply more arbitrary parametrization than the scheme we put forward, where e.g. intra-cluster variability is learned. Furthermore, their scheme is not incremental, but works in batch mode. In our view, incrementality is necessary to always keep the collection organized without user needing to think about it. Running on a mobile phone as a permanent background task with low priority, the computational demand of our technique is then far less than, for instance, real-time video codecs currently running on such platforms.

3 Proposed technique

We formulate the recovery of the image collection hierarchical spatial and temporal structures as an unsupervised classification task. Distinct hierarchical classifications are built for time and space, respectively from temporal and geolocation data. The technique exposed in this paper is used for both series, almost identically. The end-user could switch between these two classifications to browse his collection, according to what better suits his/her memory or leads to faster browsing. Alternatively, a companion paper [18] focuses on the combination of (single-scale, not hierarchical) spatial and temporal partitions into an hybrid geo-temporal representation.

3.1 Modelling and optimality criterion

We opt for the mixture model framework, in which probabilistic models are associated both to classes and data-to-class assignments. This very latter point makes it attractive for the incremental nature of the task, since data-to-class assignments may evolve in a flexible way as new data is made available.

The data D (either location (x, y) or time t stamps) is assumed to be drawn from a random Gaussian mixture process with probability density :

$$p(D) = \sum_{k=1}^{K} p_k . \mathcal{N}(D|\mu_k, \Sigma_k), \tag{1}$$

where the probabilities p_k are the mixing proportions and $\mathcal{N}(D|\mu, \Sigma)$ denotes a Gaussian distribution with mean μ and covariance Σ.

With mixture model modelling, a criterion for fair comparison between clustering hypotheses that might have different number of classes consists in comparing their integrated completed likelihoods (ICL) [3]. In contrast with the BIC criterion derived from the marginal likelihood of the data, this criterion optimizes the joint likelihood of the data D and the unobserved data-to-model assignments Z. The introduction of the latter variable accounts for the goal of discovering the hidden structure in the data. Given a clustering hypothesis H_K, the criterion is defined as :

$$p(D, Z|H_K) = \int p(D, Z|\Theta_K, H_K) p(\Theta_K|H_K) d\Theta_K \tag{2}$$

where $\Theta_K = (\theta_1, \theta_2, \ldots, \theta_K)$ is a parameter vector for H_K and $\theta_i = (p_i, \mu_i, \Sigma_i)$, $1 \leq i \leq K$. Practical computations exploit a BIC-like approximation for (2), expressed by :

$$ICL = -ML + \frac{\nu_K log(n)}{2} - \sum_{k=1}^{K} \sum_{i=1}^{n} t_{ik} \cdot log(t_{ik}) \tag{3}$$

where ML is the maximized mixture loglikelihood, ν_K is the number of independent parameters in the model with K components, n is the number of data elements and t_{ik} is the posterior probability for an observation i of originating from cluster k. These t_{ik} are in fact expectation values of the binary assignment random variables z_{ik}. In practice, they are supplied by the E step of the EM algorithm, described further below.

Expression (3) is a self-explanatory variation on the BIC criterion : the additional entropic term on the right favours well-separated clusters [3]. As a practical benefit, this increases the ability of this criterion to identify correctly non-Gaussian clusters, which our goal frequently encounters. Besides, it involves only light computation, compared to alternative mixture models (eg. mixture of Student densities [4]).

Further, it is frequent that a cluster is assigned a little amount of data, leading to a poor estimate of its empirical covariance. We deal with this issue by introducing, in the M-step, regularized covariance estimates, computed

Algorithm 1 ICL Optimisation

1.Add one datum to the data set
Run the EM algorithm, initialized by the model obtained from the present algorithm on the previous data element.
Let ICL_1 denote the value of ICL obtained at convergence and $\mathcal{M}_1$ the corresponding model.
2.Split phase:
Rank components for tentative splitting according to criterion (5).
for all the first α components, in order of decreasing entropy (typically, $\alpha = 5$) **do**
  - operate a tentative split, update $\mathcal{M}_1$ consequently.
  - run of the EM algorithm until convergence, leading to a model $\mathcal{M}_2$ with a value of ICL_2 for criterion (3).
  if $ICL_2 < ICL_1$ **then**
    $\mathcal{M}_1 \leftarrow \mathcal{M}_2$, $ICL_1 \leftarrow ICL_2$ and go back to the beginning of step 2
  end if
end for
3.Merge phase:
Rank component pairs for merging according to criterion (4).
for all the first α pairs, in order of increasing Malahanobis distance **do**
  - operate a tentative merge, update $\mathcal{M}_1$ consequently.
  - run of the EM algorithm until convergence, leading to a model $\mathcal{M}_3$ with a value of ICL_3 for criterion (3).
  if $ICL_3 < ICL_1$ **then**
    $\mathcal{M}_1 \leftarrow \mathcal{M}_3$, $ICL_1 \leftarrow ICL_3$ and go back to the beginning of step 3
  end if
end for

as expectations of the posterior distribution of these covariance matrices (using respectively Gamma and Wishart conjugate Bayesian priors for time (one dimensional) and location (two dimensional)).

In contrast with [17], all desirable properties of the scheme are incorporated into a single probabilistic model and optimality criterion, rather than a series of rules. As a result, relevance of partitions obtained may be evaluated numerically, as a whole, and if no or little structure in the data (i.e. clusters) exist (or can be found), on one of the axes (e.g. location) and on some portion of time, the user can be switched automatically to the other axis (e.g. time).

3.2 Optimization of the proposed criterion

Temporal tracking of a data partition involves adjusting the data-to-cluster assignments, as well as adjusting the number of groups when new data provides evidence in this sense. Using the solution obtained at time t to initialize the local optimisation of (3) at $t+1$ with an EM algorithm is overall a good idea :

- it ensures stability of the structure through which the user browsing is certainly beneficial;
- it supplies, at no extra cost, an explicit temporal link between groups that correspond over time.

Still, two major issues remain :

- this does not enable evolution of number of groups;
- the data stream cannot be modelled as a series of *independent* samples from a fixed probability density, in contrast with many applicative settings. As a result, the optimization hypersurface is rather shaky over time and poor local minima are often obtained if a classical conservative EM-only update is used.

A joint solution to these two issues is proposed. Briefly stated, by evaluating and (possible) applying joint split & merges among current clusters, semi-local jumps in the search space are attempted in the search space. Our approach differs significantly from the closest work [22], which does not deal with incrementality and keeps the number of groups constant. We alternate this phase (semi-local) with EM runs (local), until convergence. The process is "well-behaved", as all steps attempt to decrease the same criterion, and serves two purposes : it avoids many local minima and practically enables evolution of the number of clusters over time.

The proposed procedure is sketched in Algorithm.1 and we detail hereunder its main aspects.

Split and merge criteria:

Because of the high number of split and merge possibilities, candidates operations should first be ranked. Criteria for this are proposed in [22], but they are not suitable for small samples. For example, if components under comparison that have a single observation each, they are not deemed good candidates for a merge. In our context, this configuration is frequently encountered. Alternatively, we propose the use of the following discrepancy measure, based on the Malahanobis distance, to compare components:

$$J_{merge}(i,j,\Theta) = min\{d(\mu_i,\Sigma_i,\mu_j), \\ d(\mu_j,\Sigma_j,\mu_i)\} \tag{4}$$

where $d(\mu_j,\Sigma_j,\mu_i) = (\mu_i-\mu_j)^T \cdot \Sigma_j^{-1} \cdot (\mu_i-\mu_j)$.

Our split criterion is based on an entropic feature characterising each component and defined by (5). A high value for $J_{split}(k,\Theta)$ indeed indicates that the corresponding component does not fit well its associated data, or that another model also fits this data quite well.

$$J_{split}(k,\Theta) = -\sum_{i=1}^{n} t_{ik} \cdot log(t_{ik}) \tag{5}$$

Initialization of the new parameters :

In case of a merge, parameter initialization for the new model $\theta_{i'}$ resulting from the merge of two components parametrized by θ_i and θ_j is defined as follows :

$$p_{i'} = p_i + p_j \quad \text{and} \quad [\mu_{i'} \quad \Sigma_{i'}]^T = \frac{p_i[\mu_i \quad \Sigma_i]^T + p_j[\mu_j \quad \Sigma_j]^T}{p_i + p_j} \tag{6}$$

In case of a split of component θ_k into two components $\theta_{j'}$ and $\theta_{k'}$ relies on the following initializations :

$$p_{j'} = p_{k'} = \frac{p_k}{2}, \quad \mu_{j'} = \mu_k + \epsilon, \quad \mu_{k'} = \mu_k - \epsilon \tag{7}$$

$$\Sigma_{j'} = \Sigma_{k'} = det(\Sigma_k)^{(1/d)}/I_d \tag{8}$$

where ϵ is a small vector colinear to the eigenvector associated to the largest eigenvalue of Σ_k, $det(\Sigma)$ denotes the determinant of Σ and I_d the d-dimensional identity matrix ($d = 1$ or 2 respectively for the time and geolocation data).

Overall optimization procedure

As a new data element streams in, the incremental algorithm first attempts several splits of components, followed by several merges, and finally local EM loops. If this globally improves the ICL criterion, the search move is retained. In such a case, the list of candidates for split or merge is recomputed and the procedure loops (generally, 2 to 5 times).

Overall, the proposed approach attempts a trade-off between the ability of the scheme to scan potentially good parts of the search space and computational load. Let us make the point that, as an iterative scheme, its practical cost is tightly related to amount of structural re-organization occurring within the data set, which is usual small.

3.3 Building incrementally a hierarchy of mixture models

Main principles

When browsing in a collection of hundreds of pictures, it can be useful to complement the temporal or geolocation-based groups found, by a hierarchy formed of several levels. Efficient definition or usage of the data management system on a mobile platform can also benefit from a hierarchical organization. This section describes how we propose to build a hierarchy of mixture models, in which the technique proposed in the previous sections provides the finest-grain level. Like its basis layer, the proposed tree structure is built incrementally. Furthermore, the number of levels evolves as data streaming in provides evidence for this need.

Hierarchical mixture model-based clustering was proposed in [7], but in a batch version and for building binary trees. Our contribution may be viewed in several ways :

Algorithm 2 Hierarchical incremental algorithm

Initialization: the current node q is the root of the tree. We add the new data element *new* in the image collection.

if $c_q = \emptyset$ **then**

1. Add *new* to q and operate tentatives splits to get a finer classification of q. Add a new son $\in c_q$ for each obtained components. Go to step 6.

else

2.1. Retrieve the mixture model m associated to c_q.

2.2. Add the data element *new* in m and apply our ICL optimization (algorithm 1). The number of splits is limited to one.

if *new* is associated to an unchanged component $q' \in c_q$ **then**

3. $q = q'$ and go back to step 1.

end if

if *new* is associated to a new component q' ($q' \notin c_q$ and the set of changed component is empty) **then**

4. Add the new node $q' \in c_q$, including the parameters of q' and the image subset {new}.

end if

if *new* is associated to a changed component **then**

5.1. Retrieve the model m' associated to the leaves of changed components;

5.2. Update the model m' with our ICL optimization procedure (algorithm 1);

5.3. Build the subtree t from the components of the model m' and the unchanged component $\in c_q$ with [7];

5.4. Optimize the subtree t selecting the levels which present an optimum of the ICL criterion;

5.5. Replace the subtree of root q by t.

end if

end if

6. End of the update.

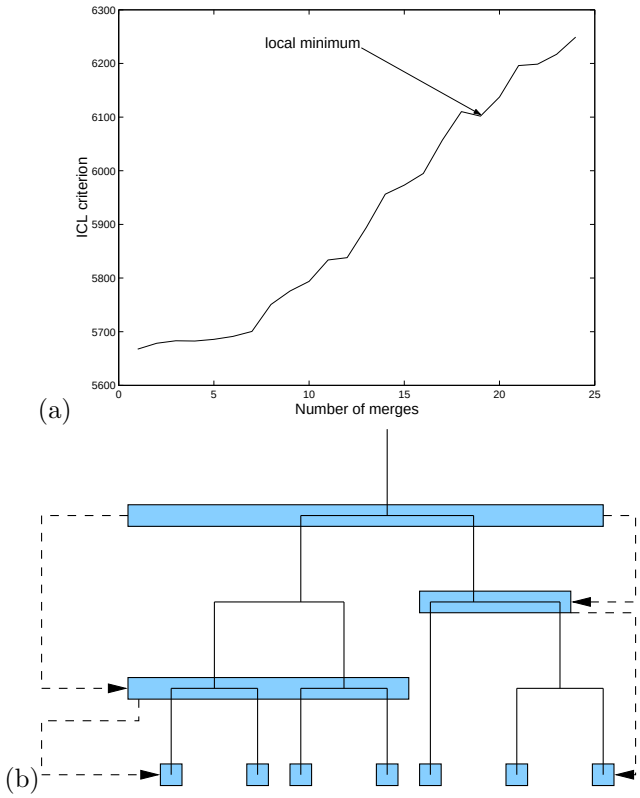

Figure 1: Selection of levels corresponding to local optima of the ICL criterion: (a) the optimal ICL criterion found at each level of the binary tree represented on (b) is plotted. The grey rectangles indicate the corresponding selection of partitions. Once an optimum is found at a level q, we search for another local optima in each subtree from q. 'Local' minima here is to be interpreted as follows : both slightly coarser and slightly finer partitions are worse, in the ICL sense.

- by creating and maintaining a view on it that is a tree containing only selected levels from the binary tree, the nodes on this view having hence a variable number of children (fig. 1). We build a binary tree, but then operate a selection among nodes, trying to avoid uninteresting and strongly redundant partitions. The ICL criterion again provides a consistent solution to the issue. Figure 1 describes this process of level selection. Proceeding from root to leaf, we search for 'local minima' in ICL, in the set of optimal partitions found at each level of the binary tree. Indeed, should there be a marked hierarchy, these local minima correspond to plausible clustering hypotheses;

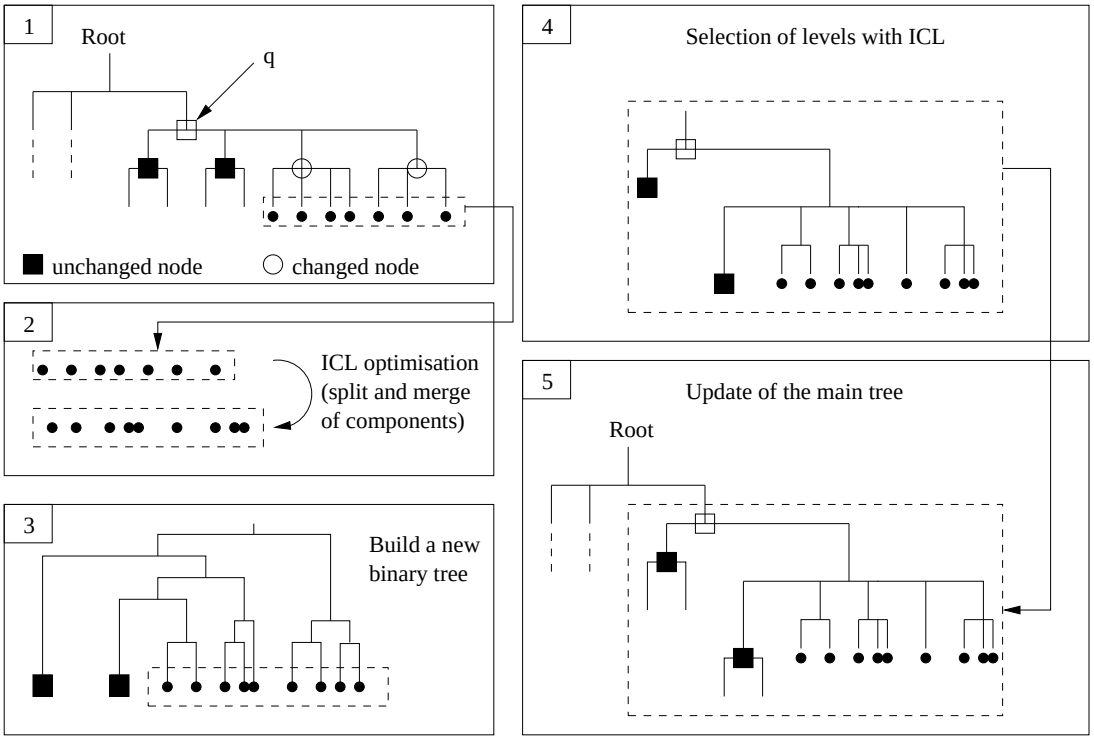

Figure 2: Update of a subtree (step 3 of our incremental hierarchical algorithm): We add the new data by the root, retrieve the model associated to its children and apply our ICL optimisation by testing splits and merges of components. In this example, no change appears and the new data element is affected to the unchanged component q. The update is then propagated to this node. Fig.1 presents the changed and unchanged nodes after the process of our ICL optimisation. We suppose here that the new data is associated to a changed component. We then retrieve the leaves of the changed nodes and applied our ICL optimisation (fig.2). We re-build a binary tree from the updated leaves and the unchanged nodes $\in c_q$ (fig.3) and select the relevant levels based on the ICL criterion. Fig.4 presents the obtained new subtree. Finally the main initial tree is updated with the subtree. Note that the children of the unchanged nodes $\in c_q$ are kept.

- by introducing a new procedure to update the hierarchy, as detailed in the next paragraph.

Procedure for updating the tree

The main idea is to propagate new data from the root to a leaf, updating each level of the hierarchy and re-organizing from scratch only sub-trees where

structural changes appear to be needed. As data proceeds from root to leaf and at each level, the incremental classification technique adjusts the parameters of the models and update data-to-model assignments (as described in section 3.2).

The scheme, when operating at other-than-leaf level, should also cope with a practical issue : it should let the structure evolve in a flexible way (including splits and merges), but not slide towards local minima corresponding to partitions already existing at finer levels of the hierarchy. This is dealt with using the following procedure. First, let us defined a node as 'changed' if the set of data assigned to it (in the Maximum a Posteriori sense and not counting the last datum) has changed after EM updates following introduction of a new data element in the scheme. The process to update a node q consists in (i) retrieving the mixture model associated to the set of its children (denoted c_q) (ii) applying our ICL optimisation, but limiting the number of splits to one, in order to avoid drifting to a partition already existing at finer levels of the hierarchy. This step determines whether introducing this last data element implies re-structuring of the updated node. According to the modification, we apply one of the following rules:

1. if the new data is associated to an 'unchanged' component , the subtree under this component is simply updated, propagating EM runs down the tree;

2. if the new data is associated to a new component, and provided no other component is changed, this corresponds to a broadening of the current level of the hierarchy;

3. if the new data is associated to a 'changed' component, this implies more important re-structuring of the data. The different steps are then:
   - select the leaves of all the changed components;
   - build a new subtree t using the technique described in [7] from the selected leaves and the unchanged components that belong to c_q;
   - optimize the tree to select a subset of levels selection (as in fig.1);
   - update the main tree with the new subtree;

The proposed procedure is detailed in Algorithm.2 and Fig.2 shows an example of an update.

A central property of the proposed technique is that the precise topology of the tree is automatically determined from the data, rather than defined by arbitrary thresholds. While the structure should be able to evolve over time, computational cost is kept low by introducing new data and updating only appropriate sections of the tree, in a top-down fashion.

4 Experimental results

We report here spatial and temporal hierarchical structures obtained using our technique, on a real personal image collection composed of 721 pictures taken

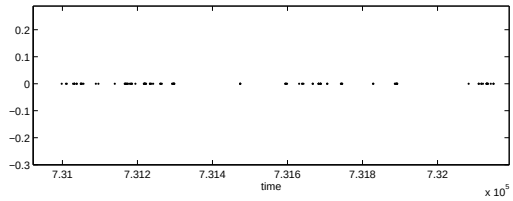

Figure 3: Real scenario: temporal metadata of the image collection. The dots represent the temporal metadata.

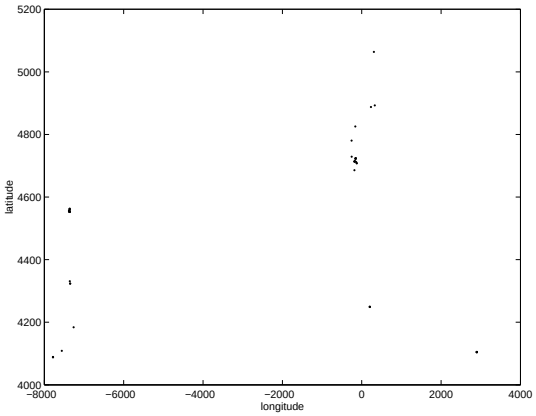

Figure 4: Real scenario: spatial metadata of the image collection. The dots represent the spatial metadata. Let us notice that the data generally forms very compact clusters, at least at the finest-grain level.

along 3 years. The user took pictures in France essentially, in USA and Canada. Time meta-data were directly included by the camera in each image (EXIF format) and the location was added manually based on the real location. Figures 3 and 4 present respectively the temporal and spatial metadata of the collection.

We first examine the temporal structure obtained over the whole collection. Figure 5 shows intermediate structures after 300, 400, 600 and 721 images. The topology of the final obtained tree is provided in Figure 5(d). Finally, selected zooms displaying the structure superimposed on the original data are depicted in Figure 6.

The tree obtained is composed of 4 levels and is well-balanced (clearly, this depends both on the data and the classification technique, but is anyway a good property for browsing). The number of children per node varies from 2 to 14. We notice on fig. 5 that our classification extends in depth and width as new data are added. All the trees present similarities (dashed squares represent the similar sections). The stability of the trees all along the classification is correct. Only a minority of images imply serious restructuring of the tree, and hence the overall computational cost grows almost linearly with the number of images.

Figure 6 presents partitions obtained at various levels of our tree. Figure 6(a) shows the coarsest level. All components are well delimited, providing relevant summaries of the collection. Components 2,3 and 4 on Figure 6(a) are respectively detailed in Figure 6(d), (b) and (c). Children of the components 4 and 3 provide well-defined partitions since all the temporal gap are correctly emphasized. For component 2, meaningful temporal episode are found but we notice over-segmentation, as groups 2.9 and 2.10, certainly due to a larger evidence of small sample associated to one component.

First experiments consist in classifying directly all the spatial metadata and provide poor results due to their characteristics. The user often took pictures in a same place, involving compact clusters with many data concentrated in one point. We obtained just one level with compact clusters for each location. Due to the data structure, our hierarchical algorithm failed to find coarser levels.
To prevent from this kind of configuration, we summarize the spatial metadata by keeping just one value of each distinct coordinate (x, y). So that one coordinate (x, y) can represent the location of several images. This summarization process provides 135 distinct locations to classify.

Figure 7 and 8 show respectively the hierarchical spatial classification and examples of obtained partitions at different levels.

The tree obtained is composed of 3 levels and the number of children per node is moderate, varying from 2 to 6. The coarser level is presented in Figure 8(a), and Figure 8(b) provides a zoom on components 1, 2 and 3. Components are well-distinct and compact due to the characteristic of the data. Our optimization algorithm have a tendency to regroup isolated data together, as shown with component 3. This aspect can nevertheless be relevant for a browsing perspective. Children of component 2 and 7 are respectively presented in fig.8(d) and (c). Both obtained partitions are quite relevant since the different main locations are found.

To evaluate practically the obtained hierarchical classification, we use the same heuristic as in [17] which are the precision and recall for the detected event boundaries:

$$precision = \frac{\text{correctly detected boundaries}}{\text{total number of detected boundaries}} \tag{9}$$

$$recall = \frac{\text{correctly detected boundaries}}{\text{total number of ground truth boundaries}} \tag{10}$$

The user manually finds 58 events in its collection. Note that he generally regroups holiday pictures or successive occasional events in a same component (both taken on several weeks). To compare with our result, we retrieve all the leaves of our temporal classification to obtain our finer classification. It is composed of 107 components and we obtained 57 correct boundaries: $recall = 98\%$. We succeed to get back all the events in the collection with a very good accuracy. Since we provide a hierarchical classification, we do not compute the precision with the finer partition. We propose to check in the temporal tree if all the finer nodes are correctly regrouped in an appropriate subtree (if

all the components of the manual partition are represented by a node in our tree). We found 50 manual components correctly regrouped in distinct nodes: $precision = 86\%$. The errors is related to holidays or occasional successive events which are divided in separate leaves.

According to the user, 25 distinct locations are present in the collection. We found successfully 23 locations with their associated images, so the spatial partition succeed to emphasize the main location of the collection. Two errors remain: one leaf regroups two close locations and one location is divided into two distinct nodes. This last case is due to images taken during a stroll where the spatial data are badly structured. The obtained hierarchy is also satisfactory since all the components regroup related locations. For example, the component 2 represents all the different locations in the home city of the user while component 1 is associated to surrounding areas.

The trade-off between temporal structural flexibility and computational load can be evaluated as follows. During the process of our algorithm on the temporal data, the agglomerative algorithm regenerating a binary tree has been called 25% of the time and, for each call, concerned on average 27% of the data. For the spatial data, it was called 60% of the time and concerned on average 8% of the data (in the first iterations, the agglomerative algorithm is often called due to the lack of data stability).

5 Summary and perspectives

This paper proposes a technique to organize a personal image collection acquired from a mobile imaging device, geographically and temporally. We focused on this meta-data since this is both useful and a low-cost, technically feasible way of recovering the events that are meaningful to users. Obviously, these criteria may be examined in conjunction with e.g. visual or audio cues, but so far we believe structuring based on geo-temporal metadata remains an open topic.

The main contribution of this paper is a fully automatic technique avoiding troublesome arbitrary parametrization and to rely solely on the data (i.e. not use geographical information systems, which may be a useful complement to name the groups of pictures). Our proposal is an incremental procedure and the way to combine it with [7] (levels quality, update aspects) to provide our incremental and hierarchical algorithm.
The overall principle is that a hierarchy of mixture models is progressively built, as new data enters the system. The integrated completed likelihood criterion was used to maintain an uniform definition of partition quality over the whole scheme. A clear separation is draw between probabilistic modelling and optimization. The probabilistic nature of assignments can handle flexible re-allocation of data to clusters, and coupling local to semi-local optimization avoids most poor local minima. The scheme is also shown to be (to some extent) robust to non-Gaussianity of clusters and small samples (i.e. a group with little data). Let us point out that the computation of the incremental algorithm is by nature distributed over time (several days) as a background task, thus consum-

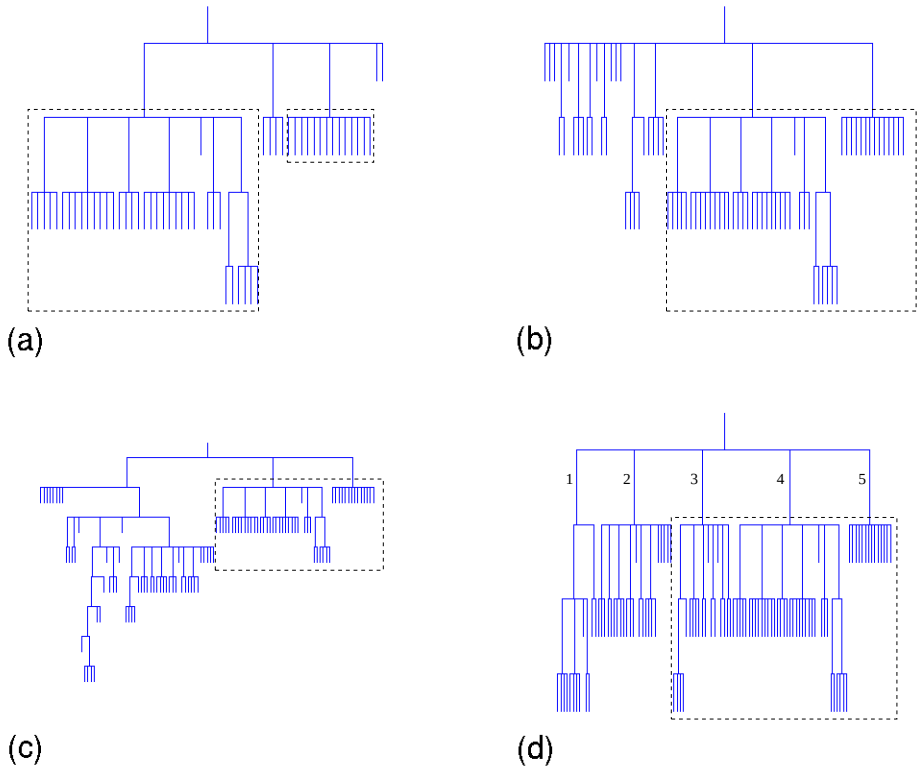

Figure 5: Real scenario: temporal hierarchical classification obtained after 300 (a), 450 (b), 600 (c), 721 (d) data. Our classification extends in depth and width as data are added. Dashed rectangle indicates the similar section of the trees all along the classification process.

ing little resources compared to other activities of mobile devices. While the focus of the paper is kept on the structuring phase, the output is dedicated to a browsing navigation interface on a mobile device. We are currently examining how to make better joint use of temporal and spatial data, given the confidence of local sections of the partitions obtained. The ICL measures may be used to this end, but the task is particularly challenging when addressing multiple scales of the hierarchy that do not necessarily correspond in time and space.

References

[1] Aris, A., Gemmel, J., and Lueder, R. Exploiting location and time for photo search

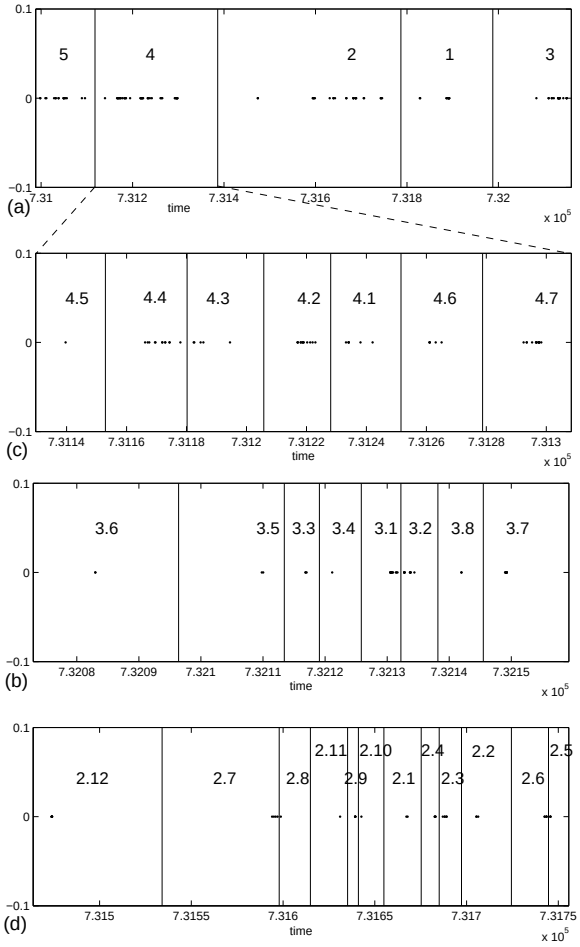

Figure 6: Real scenario: example of partitions obtained in the temporal hierarchical classification. Each number identifies the component of the associate node in figure 5(d). Solid lines represent the boundaries between components and the dots indicate the temporal data. Figure (a) represents the coarser level of our obtained temporal tree, while fig.(b), (c) and (d) respectively show the children of components 3, 4 and 2. Partitions obtained generally present distinct clusters with visually justified boundaries, although over-segmentation occasionally occurs (for example components 2.9 and 2.10).

and storytelling in MyLifeBits. Tech. Rep. MSR-TR-2004-102, Microsoft research, Sept. 2004.

[2] Ashbrook, D., and Starner, T. Learning significant locations and predicting user movement with GPS. In *IEEE Int. Symp. on Wearable Computing (ISWC'2002), Seattle, USA* (Oct. 2002), pp. 101–108.

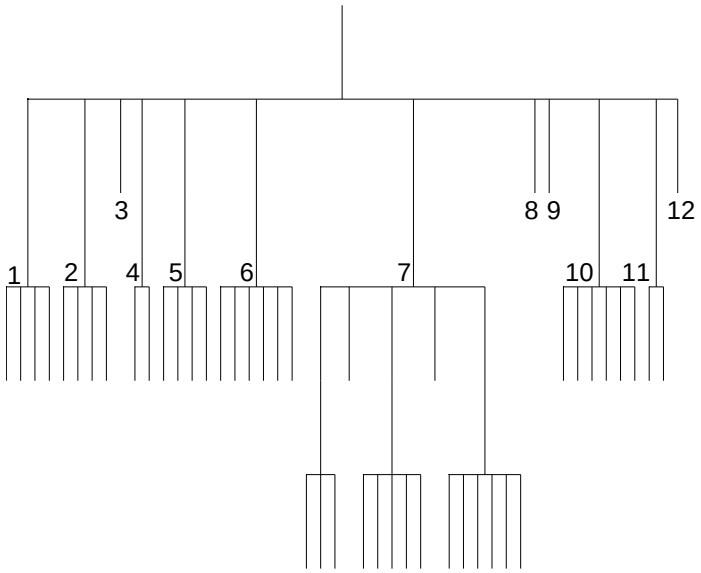

Figure 7: Real scenario: obtained spatial hierarchical classification. The obtained tree is well balanced and the number of children per nodes varies from 2 to 6. The coarser level is presented in figure fig.**??**(a). Numbers indicate a correspondence of branches to fig.**??**.

[3] Biernacki, C., Celeux, G., and Govaert, G. Assessing a mixture model for clustering with the integrated classification likelihood. In *IEEE Transaction on pattern analysis and machine intelligence* (Jul. 2000), vol. 22, pp. 719–725.

[4] Bishop, C., and Svensen, M. Robust Bayesian mixture modelling. In *Proceedings Twelfth European Symposium on Artificial Neural Networks* (Bruges, Belgium, Apr. 2004), pp. 69–74.

[5] Cooper, M., Foote, J., Girgensohn, A., and Wilcox, L. Temporal event clustering for digital photo collections. In *Proc. ACM Multimedia* (Nov. 2003), pp. 364–373.

[6] Davis, M., and Sarvas, R. Mobile media metadata for mobile imaging. In *IEEE International Conference on Multimedia and Expo (ICME 2004)* (Jun. 2004).

[7] Fraley, C. Algorithms for model-based Gaussian hierarchical clustering. *SIAM Journal on Scientific Computing 20*, 1 (1999), 270–281.

[8] Gargi, U., Deng, Y., and Tretter, D. R. Managing and searching personal photo collections. Tech. Rep. HPL-2002-67, HP Laboratories, Palo Alto, Mar. 2002.

[9] Gelgon, M. Using face detection for browsing personal slow video in a small terminal and worn camera context. In *IEEE. Int. Conf. on Image Processing (ICIP'2001)* (Thessaloniki, Greece, september 2001), pp. 1062–1065.

[10] Gelgon, M., and Tilhou., K. Structuring the personal multimedia collection of a mobile device user based on geolocation. In *IEEE Int. conf. on Multimedia and Expo (ICME'2002)* (Lausanne, Switzerland, Aug. 2002), pp. 248–252.

[11] Gemmell, J., Lueder, R., and Bell, G. The MyLifeBits lifetime store. In *Proceedings of the 2003 ACM SIGMM workshop on Experiential telepresence* (Nov. 2003), pp. 80–83.

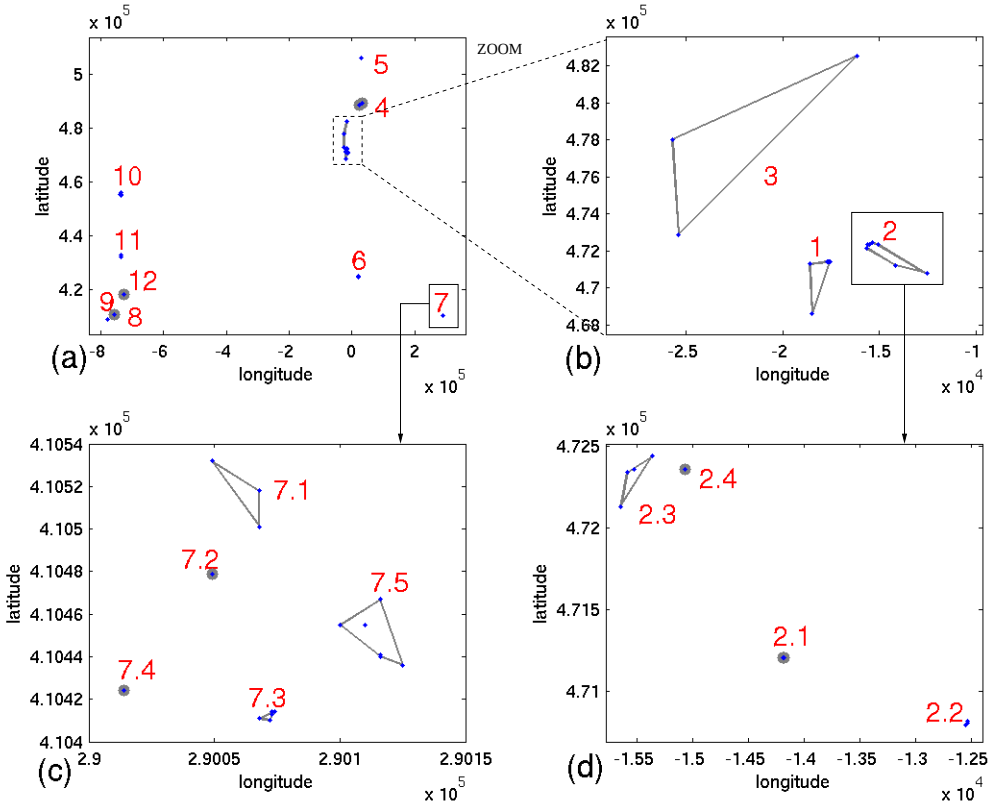

Figure 8: Real scenario: example of partitions obtained in the spatial hierarchical classification. Each number represents the components of the associate node in figure 7. The dots are the spatial metadata and thick lines are the convex hull of each component based on the maximum probability a posteriori. The arrows indicate the parental relation between the partitions. Figure (a) shows the coarser level of the classification and fig.(b) is a zoom on this partition. Fig.(c) and (d) are respectively the children of the components 7 and 2.

[12] Graham, A., Garcia-Molina, H., Paepcke, A., and Winograd, T. Time as essence for photo browsing through personal digital libraries. In *ACM Joint Conference on Digital Libraries JCDL* (Jun. 2002), pp. 326–335.

[13] Loui, A., and Savakis, A. E. Automatic image event segmentation and quality screening for albuming applications. In *IEEE Proceedings Int. Conf. on Multimedia and Expo (ICME'2000)* (New York, USA, Aug. 2000), pp. 1125–1128.

[14] Luo, J., Savakis, A., and Singhal, A. A Bayesian network-based framework for semantic image understanding. *Pattern Recognition 38*, 6 (June 2005), 919–934.

[15] M., G., Myka, A., and Yrjänäinen, J. Enhanced storing of personal content. US Patent 16660/10502275, Nokia corp., Jul. 2004.

[16] Myka, A. Nokia lifeblog - towards a truly personal multimedia information system. In *Proc. of Workshop des GI-Arbeitkreises "Mobile Datenbanken and Informationsysteme"* (Karlsruhe, Germany, Feb. 2005).

[17] Naaman, M., Song, Y. J., Paepcke, A., and Garcia-Molina, H. Automatic organization for digital photographs with geographic coordinates. In *Proc.of ACM/IEEE Conference on Digital libraries (JCDL'2004)* (Jun. 2004), pp. 53–62.

[18] Pigeau, A., and Gelgon, M. Incremental statistical geo-temporal structuring of a personal camera phone image collection. In *Proc. of Int. Conf. on Pattern Recognition (ICPR'2004)* (Cambridge, U.K., aug. 2004), pp. 878–881.

[19] Platt, J. C., and M. Czerwinski, B. A. F. PhotoTOC: Automatic clustering for browsing personal photographs. Tech. Rep. MSR-TR-2002-17, Microsoft Research, Feb. 2002.

[20] Rodden, K. How do people manage their digital photographs? In *ACM Conference on Human Factors in Computing Systems* (Fort Lauderdale, Apr. 2003), pp. 409 – 416.

[21] Toyama, K., Logan, R., Roseway, A., and Anandan, P. Geographic location tags on digital images. In *Proc. of ACM conf. on Multimedia* (Berkeley, CA, USA, Nov. 2003), pp. 156–166.

[22] Ueda, N., Nakano, R., Gharhamani, Z., and Hinton, G. SMEM algorithm for mixture models. *Neural computation 12*, 9 (2000), 2109–2128.

[23] Vermaak, J. Perez, P., and Gangnet, M. Rapid summarization and browsing of video sequences. In *Proc. of British Machine Vision Conference (BVMC'2002)* (Cardiff, U.K., Sept. 2002), pp. 145–151.

[24] Viola, P., and Jones, M. Robust real-time object detection. In *Proc. of Int. Workshop on Statistical and Computational Theories of Vision - Modeling, Learning, Computing, Sampling (with ICCV'2001)* (Vancouver, Jul. 2001).

[25] Wagenaar, W. My memory : a study of autobiographical memory over six years. *Cognitive psychology 18* (1986), 225–252.

[26] Wilhelm, A., Takhteyev, Y., Sarvas, R., Van House, N., and Davis, M. Photo annotation on a camera phone. In *Proc. of ACM Computer Human Interaction (CHI'2004)* (Vienna, Austria, Apr. 2004), pp. 234–238.