Fast Approximate Energy Minimization with Label Costs

Andrew Delong* Anton Osokin† Hossam N. Isack* Yuri Boykov*

*Department of Computer Science, University of Western Ontario

†Department of Computational Mathematics and Cybernetics, Moscow State University

Abstract

The α-expansion algorithm [4] has had a significant impact in computer vision due to its generality, effectiveness, and speed. Thus far it can only minimize energies that involve unary, pairwise, and specialized higher-order terms. Our main contribution is to extend α-expansion so that it can simultaneously optimize "label costs" as well. An energy with label costs can penalize a solution based on the set of labels that appear in it. The simplest special case is to penalize the number of labels in the solution.

Our energy is quite general, and we prove optimality bounds for our algorithm. A natural application of label costs is multi-model fitting, and we demonstrate several such applications in vision: homography detection, motion segmentation, and unsupervised image segmentation. Our C++/MATLAB implementation is publicly available.

1. Some Useful Regularization Energies

In a labeling problem we are given a set of observations $\mathcal{P}$ (pixels, features, data points) and a set of labels $\mathcal{L}$ (categories, geometric models, disparities). The goal is to assign each observation $p \in \mathcal{P}$ a label $f_p \in \mathcal{L}$ such that the joint labeling f minimizes some objective function $E(f)$.

Most labeling problems in computer vision are ill-posed and in need of regularization, but the most useful regularizers often make the problem NP-hard. Our work is about how to effectively optimize two such regularizers: a preference for fewer labels in the solution, and a preference for spatial smoothness. Figure 1 suggests how these criteria cooperate to give clean results. Surprisingly, there is no good algorithm to optimize their combination. Our main contribution is a way to simultaneously optimize both of these criteria inside the powerful α-expansion algorithm [4].

Label costs. Start from a basic (unregularized) energy $E(f) = \sum_p D_p(f_p)$, where optimal f_p can each be determined independently from the 'data costs'. Suppose, however, that we wish to explain the observations using as few unique labels as necessary. We can introduce *label costs* into $E(f)$ to penalize each unique label that appears in f:

$$E(f) = \sum_{p \in \mathcal{P}} D_p(f_p) + \sum_{l \in \mathcal{L}} h_l \cdot \delta_l(f) \qquad (1)$$

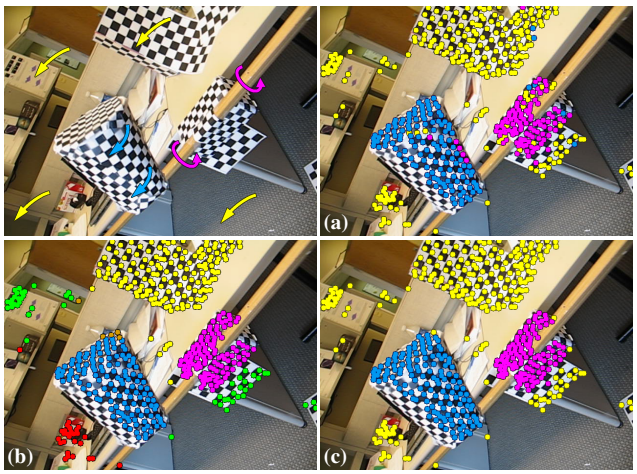

Figure 1. Motion segmentation on the 1RT2RCR sequence [26]. Energy (1) finds 3 dominant motions (a) but labels many points incorrectly. Energy (2) gives coherent segmentations (b) but finds redundant motions. Our energy combines the best of both (c).

where h_l is the non-negative label cost of label l, and $\delta_l(\cdot)$ is the corresponding indicator function

$$\delta_l(f) \stackrel{\text{def}}{=} \begin{cases} 1 & \exists p : f_p = l \\ 0 & \text{otherwise.} \end{cases}$$

Energy (1) balances data costs against label costs in a formulation equivalent to the well-studied *uncapacitated facility location* (UFL) problem. Li [19] recently posed multi-model fitting in terms of UFL. For multi-model fitting, where each label corresponds to a candidate model, label costs penalize overly-complex models, preferring to explain the data with fewer, cheaper labels (see Figure 1a).

Smooth costs. Spatial smoothness is a standard regularizer in computer vision. The idea here is that groups of observations are often known *a priori* to be positively correlated, and should thus be encouraged to have similar labels. Neighbouring image pixels are a classic example of this. Such pairwise priors can be expressed by the energy

$$E(f) = \sum_{p \in \mathcal{P}} D_p(f_p) + \sum_{pq \in \mathcal{N}} V_{pq}(f_p, f_q) \qquad (2)$$

*† The authors assert equal contribution and thus joint first authorship.

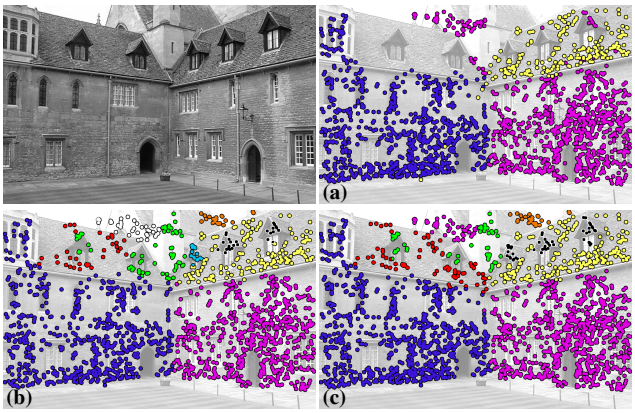

Figure 2. Planar homography detection on VGG (Oxford) Merton College 1 image (right view). Energy (1) finds reasonable parameters for only the strongest 3 models shown in (a), and still assigns a few incorret labels. Energy (2) finds reasonable clusters (b) but fits 9 models, some of which are redundant (nearly co-planar). Our energy ($\star$) finds both good parameters and labels (c) for 7 models.

where each V_{pq} penalizes $f_p \neq f_q$ in some manner. If each V_{pq} defines a metric, then minimizing (2) is known as the *metric labeling* problem [4] and can be optimized effectively with the α-expansion algorithm.

This regularizer prefers coherent segmentations, but has no incentive to combine non-adjacent segments and thus a tendency to suggest redundant labels in multi-model fitting (see Figure 1b). Still, spatial smoothness priors are important for a wide array of vision applications.

Our combined energy. We propose a discrete energy that essentially combines the UFL and metric labeling problems.

$$E(f) = \overbrace{\sum_{p\in\mathcal{P}} D_p(f_p)}^{\text{data cost}} + \overbrace{\sum_{pq\in\mathcal{N}} V_{pq}(f_p,f_q)}^{\text{smooth cost}} + \overbrace{\sum_{L\subseteq\mathcal{L}} h_L\cdot\delta_L(f)}^{\text{label cost}} \quad (\star)$$

where the indicator function $\delta_L(\cdot)$ is now defined on label subset L as

$$\delta_L(f) \stackrel{\text{def}}{=} \begin{cases} 1 & \exists p : f_p \in L \\ 0 & \text{otherwise.} \end{cases}$$

Our energy actually makes a slight generalization from label costs to label *subset* costs h_L, but one can imagine simple per-label costs h_l throughout for simplicity.

Energy ($\star$) balances two demonstrably important regularizers, as suggested by Figure 1c. Figures 2 and 3 show other vision applications where our combined energy simply makes sense. Section 2 presents our extension to α-expansion and corresponding optimality bounds. Sections 3 and 4 explain how our energy can be effective as a multi-model fitting framework. See Section 5 for discussion and possible extensions, and see [8] for relation to standard *expectation maximization* (EM) and K-means formulations.

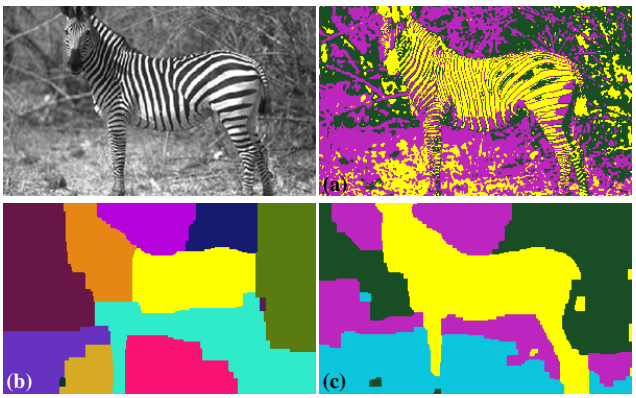

Figure 3. Unsupervised segmentation using histogram models. Energy (1) clusters in colour space, so segments (a) are incoherent. Energy (2) clusters over pixels and must either over-segment or over-smooth (b), just as in [27]. Our energy ($\star$) balances these criteria (c) and corresponds to Zhu & Yuille [28] for segmentation.

2. Fast Algorithms to Minimize ($\star$)

Our main technical contribution is to extend the well-known α-expansion algorithm [4] to incorporate label costs at each expansion (Section 2.1) and prove new optimality guarantees (Section 2.2). Section 2.3 reviews known results for the 'easy' case (1) with only data and per-label costs.

2.1. Expansion moves with label costs

Since minimizing energy ($\star$) is NP-hard for $|\mathcal{L}| \geq 3$, the α-expansion algorithm [4] iteratively 'moves' from some current labeling f' to a better one until convergence. Specifically, at each step, some label $\alpha \in \mathcal{L}$ is chosen and variables f_p are simultaneously given a *binary* choice to either stay as $f_p = f'_p$ or switch to $f_p = \alpha$. This binary step is called *expansion* because only the α label can grow and, if each V_{pq} satisfies a simple condition, the best possible expansion is computed by a single graph cut.

Let $f = \{f_1, \ldots, f_n\}$ and let f^α denote any feasible α-expansion w.r.t. current labeling f'. The possible labelings f^α can be expressed one-to-one with binary indicator variables $\mathbf{x} = \{x_1, \ldots, x_n\}$ by defining

$$\begin{array}{rcl} x_p = 0 & \iff & f^\alpha_p = f'_p \\ x_p = 1 & \iff & f^\alpha_p = \alpha. \end{array} \quad (3)$$

Let $E^\alpha(\mathbf{x})$ be the energy corresponding to encoding (3) relative to f'. The α-expansion algorithm computes an optimum $\mathbf{x}^*$, and thereby f^α, by a single graph cut.

For example, suppose energy $E(f)$ is such that the optimal expansion w.r.t labeling f' is f^α:

$$f' = \boxed{\beta}\boxed{\alpha}\boxed{\gamma}\boxed{\gamma}\boxed{\beta}\boxed{\beta} \;\rightarrow\; \begin{array}{l} \boxed{\alpha}\boxed{\alpha}\boxed{\alpha}\boxed{\gamma}\boxed{\beta}\boxed{\beta} = f^\alpha \\ \boxed{1}\boxed{\underline{1}}\boxed{1}\boxed{0}\boxed{0}\boxed{0} = \mathbf{x}^* \end{array} \quad (4)$$

where $\underline{1}$ means x_2 is fixed to 1. Here only f_1 and f_3 changed to label α while the rest preferred to keep their labels. The α-expansion algorithm iterates the above binary step until finally $E^\alpha(\mathbf{x}') = E^\alpha(\mathbf{x}^*)$ for all $\alpha \in \mathcal{L}$.

Encoding label costs. The energy in example (4) was such that f_5 and f_6 preferred to stay as label β rather than switch to α. Suppose we want to introduce a cost $h_\beta > 0$ that is added to $E(f)$ if and only if there exists some $f_p = \beta$. This would encourage label α to absorb the entire region that β occupies in f'. If h_β is large enough, the optimal α-expansion move would also change f_5 and f_6:

$$f' = \begin{array}{|c|c|c|c|c|c|}\hline \beta & \alpha & \gamma & \gamma & \beta & \beta \\ \hline \end{array} \;\rightarrow\; \begin{array}{|c|c|c|c|c|c|}\hline \alpha & \alpha & \alpha & \gamma & \alpha & \alpha \\ \hline \end{array} = f^\alpha \qquad \begin{array}{|c|c|c|c|c|c|}\hline 1 & \underline{1} & 1 & 0 & 1 & 1 \\ \hline \end{array} = \mathbf{x}^* \tag{5}$$

Our main algorithmic contribution is a way to encode such label costs into the expansion step and thereby encourage solutions that use fewer labels.

Energy $E^\alpha(\mathbf{x})$, when expressed as a multilinear polynomial, is a sum of linear and quadratic terms over $\mathbf{x}$. For the specific example (5), we can encode cost h_β in E^α by simply adding $h_\beta - h_\beta x_1 x_5 x_6$ to the binary energy. Because this specific term is cubic and $h_\beta \geq 0$, it can be optimized by a single graph cut using the construction in [16].

To encode general label costs for arbitray $L \subseteq \mathcal{L}$ and f', we must optimize the modified expansion energy

$$E^\alpha_h(\mathbf{x}) = E^\alpha(\mathbf{x}) + \sum_{\substack{L \subseteq \mathcal{L} \\ L \cap \mathcal{L}' \neq \emptyset}} \left(h_L - h_L \prod_{p \in \mathcal{P}_L} x_p \right) + C^\alpha(\mathbf{x}) \tag{6}$$

where set $\mathcal{L}'$ contains the unique labels in the current labeling f', and set $\mathcal{P}_L = \{\, p : f'_p \in L \,\}$. Term C^α simply corrects for the case when $\alpha \notin \mathcal{L}'$ and is discussed later.

Each product term in (6) adds a higher-order clique $\mathcal{P}_L$ beyond the standard α-expansion energy $E^\alpha(\mathbf{x})$. Freedman and Drineas [10] generalized the graph construction of [16] to handle terms $c \prod_p x_p$ of arbitrary degree when $c \leq 0$. This means we can transform each product seen in (6) into a sum of quadratic and linear terms that graph cuts can still optimize globally. The transformation for a particular label subset $L \subseteq \mathcal{L}$ with $|\mathcal{P}_L| \geq 3$ is

$$-h_L \prod_{p \in \mathcal{P}_L} x_p = \min_{y_L \in \{0,1\}} h_L \left[(|\mathcal{P}_L| - 1) y_L - \sum_{p \in \mathcal{P}_L} x_p y_L \right] \tag{7}$$

where y_L is an auxiliary variable that, if $h_L > 0$, must be optimized alongside $\mathbf{x}$. Since each $x_p y_L$ term has non-positive coefficient, it can be optimized by graph cuts.

To encode the potential (7) into an s-t min-cut graph construction, we reparameterize the right-hand side such that each quadratic monomial has exactly one complemented variable (e.g. $x\bar{y}$) and non-negative coefficient (arc weight). One such reparameterization is

$$-h_L + h_L \bar{y}_L + \sum_{p \in \mathcal{P}_L} h_L \bar{x}_p y_L. \tag{8}$$

where $\bar{x}_p = 1 - x_p$. Figure 4 shows the subgraph corresponding to (8) after cancelling the constant $-h_L$.

Subgraphs of this type have been used in vision before, most notably the P^n Potts potentials of Kohli et al. [14].

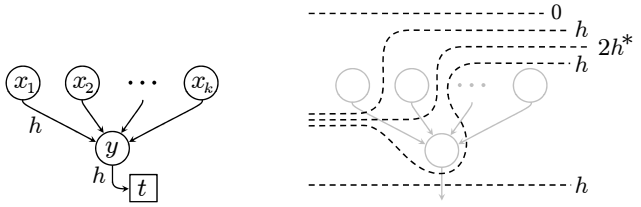

Figure 4. LEFT: Graph construction that encodes $h - hx_1x_2\cdots x_k$ when we define $x_p = 1 \Leftrightarrow p \in T$ where T is the sink side of the cut. RIGHT: In a minimum s-t cut, the subgraph contributes cost either 0 (all $x_p = 1$) or h (otherwise). A cost greater than h (e.g. $*$) cannot be minimal because setting $y = 0$ cuts only one arc.

Our indicator potentials $\delta_L(\cdot)$ are different in that, at the binary step (6), each clique $\mathcal{P}_L$ is determined *dynamically* from the current labeling f' and is not expressed as such in the original energy ($\star$). It is easy to represent a P^n Potts potential by combination of label subset cost potentials, but not the other way around. Our technical report [8] elaborates on this point and Section 5 mentions an extension to our work based on the Robust P^n Potts construction [15].

A final detail is how to handle the case when α was not used in the current labeling f'. The corrective term C^α in (6) incorporates the label costs for α itself:

$$C^\alpha(\mathbf{x}) = \sum_{\substack{L \subseteq \mathcal{L} \setminus \mathcal{L}' \\ \alpha \in L}} \left(h_L - h_L \prod_{p \in \mathcal{P}} \bar{x}_p \right). \tag{9}$$

If we find that $\mathbf{x}^* = 0$ then label α was not used in f' and it was also not worth expanding it in f^α. The term (9) can be encoded by a subgraph analogous to Figure 4, but the following is more efficient: first compute optimal $\mathbf{x}^*$ for (6) without considering C^α, then explicitly add it to $E^\alpha_h(\mathbf{x}^*)$ if $\mathbf{x}^* \neq \mathbf{0}$, and reject the expansion if the energy would increase. In fact, a similar test-and-reject step allows label costs to be trivially incorporated into α-β-swap: before accepting a swap move, test its energy against the energy when all β variables become α and vice versa.

2.2. Optimality guarantees

In what follows we assume that energy ($\star$) is configured[1] so that $D_p \geq 0$, V_{pq} is a metric [4], and thus $E(f) \geq 0$.

Theorem 1 *If f^* is a global optimum of energy ($\star$) and $\hat{f}$ is a local optimum w.r.t. α-expansion then*

$$E(\hat{f}) \;\leq\; 2cE(f^*) \;+ \sum_{L \subseteq \mathcal{L}} h_L |L| \tag{10}$$

where $\quad c = \max_{pq \in \mathcal{N}} \left(\frac{\max_{\alpha \neq \beta \in \mathcal{L}} V_{pq}(\alpha, \beta)}{\min_{\gamma \neq \zeta \in \mathcal{L}} V_{pq}(\gamma, \zeta)} \right)$

See Appendix A for the proof. The proof contains an alternate bound (26) that does not assume $D_p \geq 0$ and is tight under much more general conditions; see [8] for a discussion of tightness and local minima.

[1] Adding an arbitrary constant to $D_p(\cdot)$ or $V_{pq}(\cdot, \cdot)$ does not affect the optimal labeling, so finite costs can always be made non-negative.

The *a priori* bound (10) suggests that for large label costs the worst-case approximation is poor. The fundamental problem is that α-expansion can expand only one label at a time. It may help empirically to order the expansions in a greedy manner, but the next section describes a special case for which the greedy algorithm still yields a similar additive bound (see Section 3.5.1 of [7]). We thus do not expect much improvement unless different moves are considered.

2.3. Easy case: only per-label costs

In the absence of any smooth costs ($V_{pq} \equiv 0$) and higher-order label costs ($h_L = 0$ for $|L| > 1$) our energy reduces to a special case (1) known as the *uncapacitated facility location* (UFL) problem. This well-studied problem was recently applied for motion segmentation, first by Li [19] and then by Lazic et al. [18]. The UFL problem assigns facilities (labels) to each client (variable) such that the cost to clients is balanced against the cost of 'opening' facilities to serve them. Optimizing UFL is NP-hard by simple reduction from SET-COVER, so it is 'easier' than our full energy ($\star$) only in a practical sense.

Li optimizes the integer program corresponding to UFL by *linear programming (LP) relaxation*, then rounds fractional 'facility' variables to 0 or 1 in a straight-forward manner. Because of the heavy LP machinery, this approach is slow and affords relatively few candidate models in practice. Li implements *four* application-specific heuristics to aggressively prune candidate models "for LP's sake." Lazic et al. optimize the same energy using max-product belief propagation (BP), a message-passing algorithm.

Kuehn & Hamburger [17] proposed a natural greedy algorithm for UFL in 1963. The algorithm starts from an empty set of facilities (labels) and greedily introduces one facility at a time until no facility would decrease the overall cost. The greedy algorithm runs in $O(|\mathcal{L}|^2|\mathcal{P}|)$ time for labels $\mathcal{L}$ and observations $\mathcal{P}$. Hochbaum [12] later showed that greedy yields a $O(\log|\mathcal{P}|)$-approximation in general, and better bounds exist for special cost functions (see [23] for review). Other greedy moves have been proposed for UFL besides "open one facility at a time" (see [6, 7]).

Our C++ library implements the greedy heuristic [17] and, when smooth costs are all zero, it is 15–30 times faster than α-expansion while yielding similar energies. Indeed, "open facility α" is equivalent to expansion in this case. Note that our higher-order label costs can also be optimized greedily, but this is not standard and our bound (10) suggests the approximation may become worse.

2.4. Energy ($\star$) as an information criterion

Regularizers are useful energy terms because they can help to avoid over-fitting. In statistical model selection, various *information criteria* have been proposed to fulfil a similar role. Information criteria penalize overly-complex models, preferring to explain the data with fewer, simpler models (Occam's razor [21]).

For example, consider the well-known *Akaike information criterion* (AIC) [1]:

$$\min_{\Theta} \; -2\ln\Pr(X\,|\,\Theta) \;+\; 2|\Theta| \tag{11}$$

where Θ is a model, $\Pr(X \mid \Theta)$ is a likelihood function and $|\Theta|$ is the number of parameters in Θ that can vary. This criterion was also discussed by Torr [25] and Li [19] in the context of motion estimation.

Another well-known example is the *Bayesian information criterion* (BIC) [5, 21]:

$$\min_{\Theta} \; -2\ln\Pr(X\,|\,\Theta) \;+\; |\Theta|\cdot\ln|\mathcal{P}| \tag{12}$$

where $|\mathcal{P}|$ is the number of observations. The BIC suggests that label costs should be chosen in some proportion to the number of observations (or, in our case, the expected number of observations per model). In contrast, AIC over-fits as we add more observations from the true models. See [5] for an intuitive discussion and derivation of BIC in general, and see Torr's work [25] for insights specific to vision.

3. Working With a Continuum of Labels

Our experimental Section 4 focuses on *multi-model fitting* problems, which are the most natural applications of energy ($\star$). As was first argued in [13], energies like ($\star$) are powerful criteria for multi-model fitting in general. However, there is a technical hurdle with using combinatorial algorithms for model fitting. In such applications each label represents a specific model, including its parameter values, and the set of all labels $\mathcal{L}$ is a continuum. In line fitting, for example, $\mathcal{L} = \mathbb{R}^2$. Practically speaking, however, the combinatorial algorithms from Section 2 require a *finite* set $\mathcal{L}$ of labels (models). Below we review a technique to effectively explore the continuum of model parameters by working with a finite subset of models at any given iteration t.

PEARL Algorithm [13]

1 **propose** initial models $\mathcal{L}_0$ by random samples (as in RANSAC)
2 run α-**expansion** to compute optimal labeling f w.r.t. $\mathcal{L}_t$
3 **re-estimate** model parameters to get $\mathcal{L}_{t+1}$; $t{:=}t{+}1$; goto **2**

PEARL was the first to use regularization energies and EM-style optimization for geometric multi-model fitting. Other geometric model fitting works have used separate elements such as random sampling [25, 19] (as in RANSAC) or EM-style iteration [2], but none have combined them in a single optimization framework. The experiments in [13] show that their energy-based formulation beats many state-of-the-art algorithms in this area. In other settings (segmentation, stereo) these elements have been combined in various application-specific ways [28, 2, 22, 27].

Our paper introduces a more general energy ($\star$) and a better algorithm for the expansion step of PEARL (step **2**).

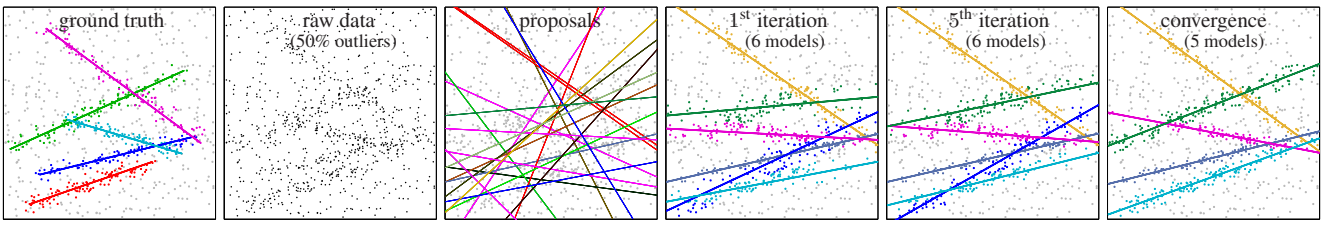

Figure 5. Re-estimation helps to align models over time. Above shows 900 raw data points with 50% generated from 5 line intervals. Random sampling proposes a list of candidate lines (we show 20 out of 100). The 1st segmentation and re-estimation corresponds to Li [19], but only the yellow line and gray line were correctly aligned. The decreasing energies in Figure 8 correspond to better alignments like the subsequent iterations above. If a model loses enough inliers during this process, it is dropped due to label cost (dark blue line).

Review of PEARL for ($\star$). Step **1** of PEARL is to propose an initial set of models $\mathcal{L}_0$. Each proposal is generated by a randomly sampling the smallest subset of data points needed to define a geometric model, exactly as in RANSAC [9]. A larger set of proposals $\mathcal{L}_0$ is more likely to contain models that approximate the true ones. Of course, $\mathcal{L}_0$ will contain many incorrect models as well, but optimizing energy ($\star$) over $\mathcal{L}_0$ (step **2**) will automatically select a small subset of labels from among the best models in $\mathcal{L}_0$.

The initial set of selected models can actually be further improved as follows. From here on, we represent model assignments by two sets of variables: segmentation variables $\{f_p\}$ that for each data point p specifies the index of a model from the finite set $\mathcal{L}_0$, and parameter variables $\{\theta_l\}$ that specify model parameters currently associated with each model index. Then, energy ($\star$) is equivalent to

$$E(f;\theta) = \sum_{p\in\mathcal{P}} D_p(f_p,\theta_{f_p}) + \sum_{pq\in\mathcal{N}} V_{pq}(f_p,f_q,\theta_{f_p},\theta_{f_q}) + \sum_{L\subseteq\mathcal{L}} h_L(\theta_L)\cdot\delta_L(f). \qquad (\star')$$

For simplicity, assume that the smoothness terms in ($\star'$) are Potts interaction potentials [4] and the third term represents simple per-label costs as in (1). Then, specific model parameters θ_l assigned to a cluster of points $\mathcal{P}_l = \{p \,|\, f_p = l\}$ only affect the first term in ($\star'$), which is a sum of unary potentials. In most cases, it is easy to compute a parameter value $\hat{\theta}_l$ that locally or even globally minimizes $\sum_{p\in\mathcal{P}_l} D_p(l,\theta_l)$. The re-estimated parameters $\{\hat{\theta}_l\}$ correspond to an improved set of labels $\mathcal{L}_1$ that reduces energy ($\star'$) for fixed segmentation f (step **3**).

Now one can re-compute segmentation f by applying the algorithms in Section 2 to energy ($\star$) over a new set of labels $\mathcal{L}_1$ (step **2** again). PEARL's re-segmentation and re-estimation steps **2** and **3** reduce the energy. Iterating these two steps generates a sequence of re-estimated models $\mathcal{L}_0, \mathcal{L}_1, \mathcal{L}_2, \ldots$ converging to a better local minima of energy ($\star$). In our experiments, convergence is typically achieved in 5–20 iterations. In most cases, iterating improves the solution quality significantly beyond the initial iteration (see Figure 8).

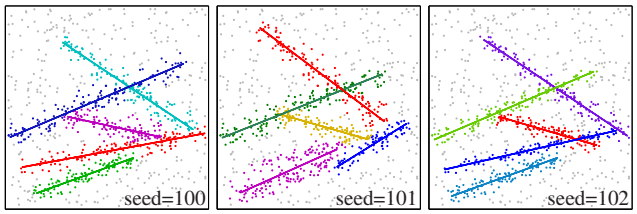

Figure 6. We can also fit line *intervals* to the raw data in Figure 5. The three results above were each computed from a different set $\mathcal{L}$ of random initial proposals. See Section 4.1 for details.

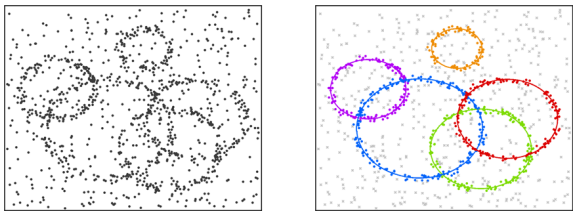

Figure 7. For multi-model fitting, each label can represent a specific model from any family (Gaussians, lines, circles...). Above shows circle-fitting by minimizing geometric error of points.

4. Applications and Experimental Setup

The experimental setup is essentially the same for each application: generate proposals via random sampling, compute initial data costs D_p, and run the iterative algorithm from Section 3. The only components that change are the application-specific D_p and regularization settings. Section 4.1 outlines the setup for basic geometric models: lines, circles, homographies, motion. Section 4.2 describes the unsupervised image segmentation setup.

4.1. Geometric multi-model fitting

Each label $l \in \mathcal{L}$ represents an instance from a specific class of geometric model (lines, homographies), and each $D_p(l)$ is computed by some class-specific measure of geometric error. The strength of per-label costs and smooth costs were tuned for each application.

Outliers. All our experiments handle outliers in a standard way: we introduce a special outlier label ϕ with $h_\phi = 0$ and $D_p(\phi) = \text{const} > 0$ manually tuned. This corresponds to a uniform distribution of outliers over the domain.

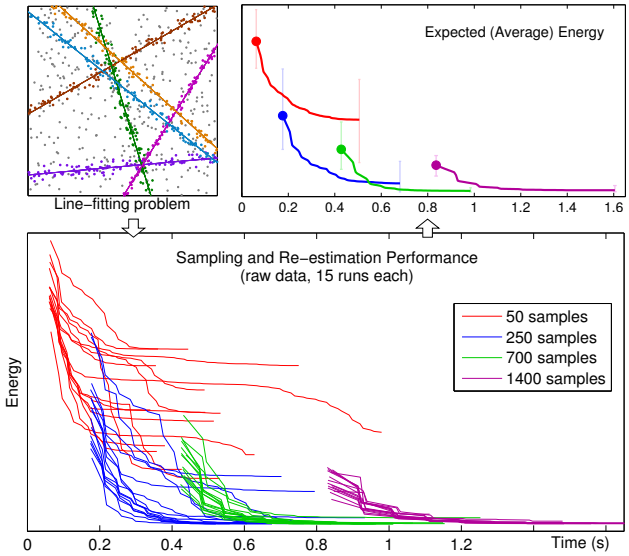

Figure 8. Energy ($\star'$) over time for a line-fitting example (1000 points, 40% outliers, 6 ground truth models). Only label cost regularization was used. Re-estimation reduces energy faster and from fewer samples. The first point ($\bullet$) in each series is taken after exactly one segmentation/re-estimation, and thus suggests the quality of Li [19] using a greedy algorithm instead of LP relaxation.

Line fitting. Our line fitting experiments are all synthetic and mainly meant to be illustrative. Data points are sampled i.i.d. from a ground-truth set of line segments (e.g. Figure 5), under reasonably similar noise; outliers are sampled uniformly. Since the data is i.i.d. we set $V_{pq} = 0$ and use the greedy algorithm from Section 2.3. Figure 5 is a typical example of our line fitting result with outliers.

In 2D each line model l has parameters $\theta_l = \{a, b, c, \sigma\}$ where $ax+by+c = 0$ defines the line and σ^2 is the variance of data; here a, b, c have been scaled such that $a^2 + b^2 = 1$. Each proposal line is generated by selecting two random points from $\mathcal{P}$, fitting a, b, c accordingly, and selecting a random initial σ based on a prior. The data cost for a 2D point $x_p = (x_p^{\mathrm{x}}, x_p^{\mathrm{y}})$ is computed w.r.t. orthogonal distance

$$D_p(l) \;=\; -\ln\Big(\tfrac{1}{\sqrt{2\pi}\sigma}\exp\Big(-\tfrac{(ax_p^{\mathrm{x}}+bx_p^{\mathrm{y}}+c)^2}{2\sigma^2}\Big)\Big)\,. \qquad (13)$$

Figure 8 shows the trend in running time as the number of random initial proposals is increased. For 1000 data points and 700 samples, convergence took .7–1.2 seconds with 50% of execution time going towards computing data costs (13) and performing re-estimation.

Line interval and circle fitting. Figure 6 shows three interval fitting results from different initial proposals. Figure 7 shows a simple circle fitting result. See [8] for details.

Homography estimation. Our setup comes directly from [13] so we give only an outline. The input comprises two (static) images related by a fundamental matrix. We first detect SIFT features [20] and do matching as a preprocessing step; these matches are our observations. Our set of neighbours $pq \in \mathcal{N}$ is determined by a Delaunay triangulation of feature positions in the first image. The models being estimated are homographies, and each proposal is generated by sampling four potential feature matches. Data costs measure the symmetric transfer error [11] of a match w.r.t a homography. Figure 2 shows a representative result.

Multi-body motion segmentation. The setup starts the same as for homography estimation, except here each model is a fundamental matrix corresponding to a rigid body motion, as in [19], and each proposal is generated by sampling eight matches. Data costs measure the squared Sampson's distance [11] of a match w.r.t. a fundamental matrix. Figure 1 shows a representative result. See [13] for details.

4.2. Image segmentation by MDL criterion

Here the models are either greyscale histograms or Gaussian mixtures in RGB space. Initial proposals were generated by sampling small patches of the input image, just like in [28, 27]. We used uniform Potts model for pairwise terms. See [8] for further details. Figures 3 and 9 show examples of our segmentations.

We formulate the problem as one of finding a *minimum description length* (MDL) representation for the image, meaning a we want to represent the image compactly, in an information-theoretic sense (see [21] for review of MDL). The MDL principle was first proposed for unsupervised segmentation by Zhu & Yuille [28], along with their *region competition* algorithm. When defined over a 2D grid of image pixels, our energy ($\star'$) can implement a discrete version of Zhu & Yuille's energy. Our algorithm is however more powerful because α-expansion makes large moves, while region competition relies on local contour evolution and explicit merging of adjacent regions.

5. Discussion

Our C++ implementation and MATLAB wrapper are available at `http://vision.csd.uwo.ca/code/`. The potential applications of our algorithm are nearly as broad as for α-expansion. Our algorithm can be applied when observations are known *a priori* to be correlated, whereas standard mixture model algorithms are designed for i.i.d. data.

We can generalize the concept of label costs by making them spatially variant. The label cost term in energy ($\star$) could actually be expressed as

$$\sum_{P\subseteq\mathcal{P}}\sum_{L\subseteq\mathcal{L}} h_L^P\cdot\delta_L(f_P) \qquad (14)$$

where the energies discussed in this paper are the special case when $h_L^P = 0$ for all clique sets $P \subsetneq \mathcal{P}$. Note that the test-and-reject approach (Section 2.1) to incorporate $C^\alpha(\cdot)$ may no longer be ideal for such 'regional' label costs.

Regional label costs may be useful when labels belong to known categories with specific priors, such as "pay a fixed

Figure 9. Unsupervised segmentation by clustering simultaneously over pixels and colour space using Gaussian mixtures (colour images) and non-parametric histograms (greyscale images). Notice we find coarser clustering on baseball than Zabih & Kolmogorov [27] without over-smoothing. For segmentation, our energy is closer to Zhu & Yuille [28] but our algorithm is more powerful than region-competition.

penalty if any label from $\{sky, cloud, sun\}$ appears in the bottom of an image." Indeed, our higher-order label costs themselves seem to be novel, both in vision and in terms of the UFL problem, and can be thought of as a specific type of co-occurrence cost.

Furthermore a binary construction based on Robust P^n Potts [15], within our expansion step, allows us to encode an arbitrary concave penalty on the number of variables taking a specific label, thus generalizing $\delta_l(\cdot)$ if needed. We leave this as future work.

Our energy is quite general but this can be a disadvantage in terms of speed. The α-expansion step runs in polynomial time for fixed number of positive h_L terms, but higher-order label costs should be used sparingly. Even the set of per-label costs $\{h_l\}$ slows down α-expansion by 40–60%, though this is still relatively fast for such difficult energies [24]. This slowdown may be because the Boykov-Kolmogorov maxflow algorithm [3] relies on heuristics that do not work well for large cliques, i.e. subgraphs of the kind in Figure 4. Even if faster algorithms can be developed, our implementation can test the merit of various energies before one invests time in specialized algorithms.

Acknowledgements We would like to thank **Fredrik Kahl** for referring us to the works of Li [19] and Vidal [26], and for suggesting motion segmentation as an application. This work was supported by NSERC (Canada) Discovery Grant R3584A02 and Russian President Grant MK-3827.2010.9.

A. Optimality proof

Proof of Theorem 1. The proof idea follows Theorem 6.1 of [4]. Let us fix some $\alpha \in \mathcal{L}$ and let

$$\mathcal{P}_\alpha \stackrel{\text{def}}{=} \left\{ p \in \mathcal{P} : f^*_p = \alpha \right\}. \tag{15}$$

We can produce a labeling f^α within one α-expansion move from $\hat{f}$ as follows:

$$f^\alpha_p = \begin{cases} \alpha & \text{if } p \in \mathcal{P}_\alpha \\ \hat{f}_p & \text{otherwise.} \end{cases} \tag{16}$$

Since $\hat{f}$ is a local optimum w.r.t. expansion moves we have

$$E(\hat{f}) \le E(f^\alpha). \tag{17}$$

Let $E(\cdot)|_{\mathcal{S}}$ denote a restriction of the summands of energy ($\star$) to only the following terms:

$$E(f)|_{\mathcal{S}} = \sum_{p \in \mathcal{S}} D_p(f_p) + \sum_{pq \in \mathcal{S}} V_{pq}(f_p, f_q).$$

We separate the unary and pairwise terms of $E(f)$ via interior, exterior, and boundary sets with respect to pixels $\mathcal{P}_\alpha$:

$$\begin{aligned} \mathcal{I}^\alpha &= \mathcal{P}_\alpha \cup \{ pq \in \mathcal{N} : p \in \mathcal{P}_\alpha, q \in \mathcal{P}_\alpha \} \\ \mathcal{O}^\alpha &= \mathcal{P} \setminus \mathcal{P}_\alpha \cup \{ pq \in \mathcal{N} : p \notin \mathcal{P}_\alpha, q \notin \mathcal{P}_\alpha \} \\ \mathcal{B}^\alpha &= \{ pq \in \mathcal{N} : p \in \mathcal{P}_\alpha, q \notin \mathcal{P}_\alpha \}. \end{aligned}$$

The following facts now hold:

$$E(f^\alpha)|_{\mathcal{I}^\alpha} = E(f^*)|_{\mathcal{I}^\alpha} \tag{18}$$

$$E(f^\alpha)|_{\mathcal{O}^\alpha} = E(\hat{f}\,)|_{\mathcal{O}^\alpha} \tag{19}$$

$$E(f^\alpha)|_{\mathcal{B}^\alpha} \le cE(f^*)|_{\mathcal{B}^\alpha}. \tag{20}$$

Equation (20) holds because for any $pq \in \mathcal{B}^\alpha$ we have $V_{pq}(f^\alpha_p, f^\alpha_q) \le cV_{pq}(f^*_p, f^*_q)$.

Let E_H denote the label cost terms of energy E. Using (18), (19) and (20) we can rewrite (17) as

$$E(\hat{f}\,)|_{\mathcal{I}^\alpha} + E(\hat{f}\,)|_{\mathcal{B}^\alpha} + E_H(\hat{f}\,) \tag{21}$$

$$\le E(f^\alpha)|_{\mathcal{I}^\alpha} + E(f^\alpha)|_{\mathcal{B}^\alpha} + E_H(f^\alpha) \tag{22}$$

$$\le E(f^*)|_{\mathcal{I}^\alpha} + cE(f^*)|_{\mathcal{B}^\alpha} + E_H(f^\alpha) \tag{23}$$

Depending on $\hat{f}$ we can bound $E_H(f^\alpha)$ by

$$E_H(f^\alpha) \le E_H(\hat{f}) + \begin{cases} \sum\limits_{\substack{L \subseteq \mathcal{L} \setminus \hat{\mathcal{L}} \\ \alpha \in L}} h_L & \text{if } \alpha \in \mathcal{L}^* \\ 0 & \text{otherwise.} \end{cases} \tag{24}$$

where sets $\mathcal{L}^*$ and $\hat{\mathcal{L}}$ contain the unique labels in f^* and $\hat{f}$ respectively.

To bound the total energy we sum expressions (21) and (23) over all labels $\alpha \in \mathcal{L}^*$ to arrive at the following:

$$\sum_{\alpha\in\mathcal{L}^*}\Big(E(\hat{f}\,)|_{\mathcal{I}^\alpha} + E(\hat{f}\,)|_{\mathcal{B}^\alpha}\Big) \tag{25}$$
$$\leq \sum_{\alpha\in\mathcal{L}^*}\Big(E(f^*)|_{\mathcal{I}^\alpha} + cE(f^*)|_{\mathcal{B}^\alpha}\Big) + \sum_{L\subseteq\mathcal{L}\setminus\hat{\mathcal{L}}} h_L|L\cap\mathcal{L}^*|.$$

Observe that, for every $pq \in \mathcal{B} = \bigcup_{\alpha\in\mathcal{L}}\mathcal{B}^\alpha$, the term $V_{pq}(\hat{f}_p, \hat{f}_q)$ appears twice on the left side of (25), once for $\alpha = \hat{f}_p^*$ and once for $\alpha = \hat{f}_q^*$. Similarly every $V(f_p^*, f_q^*)$ appears $2c$ times on the right side of (25). Therefore equation (25) can be rewritten as

$$\begin{aligned} E(\hat{f}) \;\leq\; & E(f^*) + (2c-1)E(f^*)|_{\mathcal{B}} - E(\hat{f})|_{\mathcal{B}} \\ & + E_H(\hat{f}) - E_H(f^*) + \sum_{L\subseteq\mathcal{L}\setminus\hat{\mathcal{L}}} h_L|L\cap\mathcal{L}^*|. \end{aligned} \tag{26}$$

The above inequality is a tight *a posteriori* bound on $E(\hat{f})$ w.r.t. a specific local optimum $\hat{f}$ and global optimum f^*; see [8] for worst-case local minima. Observe that

$$\begin{aligned} & E_H(\hat{f}) - E_H(f^*) \;+ \sum_{L\subseteq\mathcal{L}\setminus\hat{\mathcal{L}}} h_L|L\cap\mathcal{L}^*| \\ & = \sum_{\substack{L\subseteq\mathcal{L}\setminus\mathcal{L}^* \\ L\cap\hat{\mathcal{L}}\neq\emptyset}} h_L \;+ \sum_{\substack{L\subseteq\mathcal{L}\setminus\hat{\mathcal{L}} \\ L\cap\mathcal{L}^*\neq\emptyset}} h_L\big(|L\cap\mathcal{L}^*| - 1\big) \\ & \leq \sum_{L\subseteq\mathcal{L}} h_L|L|. \end{aligned} \tag{27}$$

Using (27) and the assumption $D_p \geq 0$ we simplify (26) to give *a priori* bound (10). ■

References

[1] H. Akaike. A new look at statistical model identification. *IEEE Trans. on Automatic Control*, 19:716–723, 1974. 4

[2] S. Birchfield and C. Tomasi. Multiway cut for stereo and motion with slanted surfaces. In *ICCV*, 1999. 4

[3] Y. Boykov and V. Kolmogorov. An Experimental Comparison of Min-Cut/Max-Flow Algorithms for Energy Minimization in Vision. *IEEE TPAMI*, 29(9):1124–1137, 2004. 7

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast Approximate Energy Minimization via Graph Cuts. *IEEE TPAMI*, 23(11):1222–1239, 2001. 1, 2, 3, 5, 7

[5] K. P. Burnham and D. R. Anderson. *Model Selection and Multimodel Inference*. Springer, 2002. 4

[6] G. Cornuejols, M. L. Fisher, and G. L. Nemhauser. Location of Bank Accounts to Optimize Float: An Analytic Study of Exact and Approximate Algorithms. *Management Science*, 23(8):789–810, 1977. 4

[7] G. Cornuejols, G. L. Nemhauser, and L. A. Wolsey. The Uncapacitated Facility Location Problem. Technical Report 605, Op. Research, Cornell University, August 1983. 4

[8] A. Delong, A. Osokin, H. N. Isack, and Y. Boykov. Fast Approximate Energy Minimization with Label Costs. Technical Report 731, Univ. of Western Ontario, Dec 2009. 2, 3, 6, 8

[9] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 5

[10] D. Freedman and P. Drineas. Energy minimization via graph cuts: settling what is possible. In *CVPR*, June 2005. 3

[11] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003. 6

[12] D. S. Hochbaum. Heuristics for the fixed cost median problem. *Mathematical Programming*, 22(1):148–162, 1982. 4

[13] H. N. Isack and Y. Boykov. Energy-based Geometric Multi-Model Fitting. Technical Report 735, University of Western Ontario, March 2010. (Submitted to IJCV). 4, 6

[14] P. Kohli, M. P. Kumar, and P. H. S. Torr. P^3 & Beyond: Solving Energies with Higher Order Cliques. In *CVPR*, 2007. 3

[15] P. Kohli, L. Ladický, and P. H. S. Torr. Robust Higher Order Potentials for Enforcing Label Consistency. *IJCV*, 82(3):302–324, 2009. 3, 7

[16] V. Kolmogorov and R. Zabih. What Energy Functions Can Be Optimized via Graph Cuts. *IEEE TPAMI*, 26(2):147–159, 2004. 3

[17] A. A. Kuehn and M. J. Hamburger. A Heuristic Program for Locating Warehouses. *Manag. Sci.*, 9(4):643–666, 1963. 4

[18] N. Lazic, I. Givoni, B. Frey, and P. Aarabi. FLoSS: Facility Location for Subspace Segmentation. In *ICCV*, 2009. 4

[19] H. Li. Two-view Motion Segmentation from Linear Programming Relaxation. In *CVPR*, 2007. 1, 4, 5, 6, 7

[20] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 60:91–110, 2004. 6

[21] D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003. 4, 6

[22] C. Rother, V. Kolmogorov, and A. Blake. GrabCut: Interactive Foreground Extraction using Iterated Graph Cuts. In *SIGGRAPH*, 2004. 4

[23] D. B. Shmoys, E. Tardos, and K. Aardal. Approximation algorithms for facility location problems (extended abstract). In *ACM STOC*, pages 265–274, 1998. 4

[24] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for Markov random fields with smoothness-based priors. *IEEE TPAMI*, 30(6):1068–1080, June 2008. 7

[25] P. H. S. Torr. Geometric Motion Segmentation and Model Selection. *Philosophical Trans. of the Royal Society A*, pages 1321–1340, 1998. 4

[26] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *CVPR*, 2007. 1, 7

[27] R. Zabih and V. Kolmogorov. Spatially Coherent Clustering with Graph Cuts. In *CVPR*, June 2004. 2, 4, 6, 7

[28] S. C. Zhu and A. L. Yuille. Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *TPAMI*, 18(9):884–900, 1996. 2, 4, 6, 7