Minimal Basis Facility Location for Subspace Segmentation

Choon-Meng Lee and Loong-Fah Cheong
ECE Department, National University of Singapore
{leechoonmeng,eleclf}@nus.edu.sg

Abstract

In contrast to the current motion segmentation paradigm that assumes independence between the motion subspaces, we approach the motion segmentation problem by seeking the parsimonious basis set that can represent the data. Our formulation explicitly looks for the overlap between subspaces in order to achieve a minimal basis representation. This parsimonious basis set is important for the performance of our model selection scheme because the sharing of basis results in savings of model complexity cost. We propose the use of affinity propagation based method to determine the number of motion. The key lies in the incorporation of a global cost model into the factor graph, serving the role of model complexity. The introduction of this global cost model requires additional message update in the factor graph. We derive an efficient update for the new messages associated with this global cost model. An important step in the use of affinity propagation is the subspace hypotheses generation. We use the row-sparse convex proxy solution as an initialization strategy. We further encourage the selection of subspace hypotheses with shared basis by integrating a discount scheme that lowers the factor graph facility cost based on shared basis. We verified the model selection and classification performance of our proposed method on both the original Hopkins 155 dataset and the more balanced Hopkins 380 dataset.

1. Introduction

We motivate our work by examining the use of spectral clustering[20][25][4] in motion segmentation. Spectral clustering has proven to be an effective and robust clustering method in the motion segmentation literature. Sparse Subspace Clustering(SSC)[6], Low Rank Representation(LRR)[18] and Linear Subspace Spectral Clustering(LSSC)[13] use spectral clustering for motion segmentation to achieve excellent results. These methods assume known number of motion when using spectral clustering. Recently, Ordered Residual Kernel(ORK)[3] and LRR extend the use of spectral clustering for model selection, based on the number of zero singular values in the normalized Laplacian. In the presence of noise, this is challenging because the singular values of the Laplacian are seldom zero. In fact, the gap between the supposed zero singular values and the non-zero singular values is often ill-defined. LRR came up with a robust thresholding operator in response to this difficulty and achieved state-of-the-art performance at 78.06%[1] for the Hopkins 155 dataset, clearly with much room for improvement. This difficulty is better understood when we look at the limitations of spectral clustering below.

The appeal of spectral clustering lies in the use of local pairwise affinity information to derive global eigenvector information for clustering. Even though the construction of the affinity matrix may involve global information, the final affinity matrix only contain local pairwise similarity measure. For example, the nuclear norm regularization that LRR uses is global in nature, but the final self representation matrix describes pairwise trajectory affinity.

In [19], the fundamental limits of spectral clustering are analyzed. The two issues raised are highly relevant in the motion segmentation context. The first concern questions if the local affinity information is sufficient for global clustering. It turns out that local information is insufficient when the data consists of clusters at different scale. The second concern calls into question the use of the first k eigenvectors to find k clusters when confronted with multi-scale and multi-density clusters.

Although these limitations were discussed in the context of classification, they carry over to model selection as well. Recall that model selection in spectral clustering is based on identifying the number of zero singular values. When the complication of multi-scale, multi-density and noise set in, the number of zero singular values is different when the Laplacian is examined at different scale. The difficulty of model selection using spectral clustering can thus be understood as ambiguity brought about by multi-scale and multi-density data clusters.

In motion segmentation, multi-scale and multi-density data clusters are very real issues that affect the performance

[1]The figure of 77.56% reported in [18] is based on 156 sequences

of spectral clustering based methods. Compared to the foreground motion, the background motion tends to contain feature points that span a larger extent of their subspace(due to the greater range of depth and (x, y) location of these points). This leads to multi-scale and multi-density data clusters.

In view of the limitations of spectral clustering, we adopt an alternative paradigm for model selection and segmentation based on global trajectory-subspace distance information. Instead of reducing it to local trajectory-trajectory affinity representation, we generate a set of subspace hypotheses and compute the distance between the trajectories and the subspace hypothesis. With this measure of affinity to subspace hypotheses, model selection is based on the affinity propagation(AP)[8] framework with a judiciously chosen global cost function.

Clearly, there are several motion segmentation works[15][3][2] that are based on trajectory-subspace distance information, but not many of them develop their work for model selection. Kernel Optimization(KO)[2] is a notable exception in that it achieves a good model selection performance. However, KO's random subspace hypotheses generation strategy is different from our work. The subsequent treatment of these subspace hypotheses is also different from our approach. KO merges these subspace hypotheses in a greedy manner, choosing the pair with the lowest kernel-target alignment at each step.

In section 3, we demonstrate how a minimal basis subspace hypotheses set can be generated by requiring the representation matrix to be jointly row sparse. Due to the convex relaxation artefact, the number of subspace hypotheses is far greater than the true number of subspaces. In section 4, we show how to incorporate a general model complexity term into the AP framework naturally and efficiently. This model complexity term is important in ensuring that the the right number of subspaces from the hypotheses set are chosen for representation. Although the subspace hypotheses set contains many overlapping subspaces, we still need to ensure the selection of those overlapping subspaces by introducing the facility cost discount scheme. We describe this discount scheme in the same section. In section 5, we verify our proposed work on the original and augmented Hopkins dataset, demonstrating a model selection performance significantly better than the state-of-the-art.

Our paper contribution is three fold. Our first contribution is in the formulation and realization of the minimal basis approach to model selection. Our method is significantly different from the current motion segmentation paradigm that uses spectral clustering. We demonstrate unequivocally the model selection strength of our proposed method.

The second contribution is the recognition, handling and leveraging of possible subspace dependencies. Whereas every current algorithm assumes subspace independence, treating the overlap as noise, our proposed work properly accounts for subspace dependencies by offering facility cost discount for shared basis. The use of these shared basis subspace for representation has important application in areas such as articulated motion and non-rigid structure from motion.

Lastly, we show how the introduction of a global facility cost function to the AP framework enables model selection with good performance while maintaining efficiency.

2. Previous work

Affinity propagation(AP) provides an interesting comparison with spectral clustering. In affinity propagation [8][11], the goal is to look for representative data points called exemplars and cluster the rest of the data points based on similarity to the exemplars. The number of clusters is not specified in AP. Instead, the number of clusters is controlled by the preference value assigned to each data point. The preference value can be regarded as the importance of a data in terms of becoming an exemplar. If a data point has a high preference value, then it has a better chance of becoming an exemplar. As an illustration, suppose the preference value is common across all the data points. If this common preference value is large, a larger number of clusters will emerge. Vice versa, a smaller common preference value will result in a smaller number of clusters. The affinity propagation clustering method has been applied to image categorization[5] and extended to motion segmentation in FLoSS(Facility Location for Subspace Segmentation)[15] and UFLP(Uncapacitated Facility Location Problem)[14]).

In FLoSS and UFLP, motion segmentation is formulated as an instance of the facility location(FL) problem. FL is known to be NP hard and hence difficult to solve. An approximate solution for FL can be found by performing maximum-a-posteri(MAP) inference in a probabilistic graphical model. In FLoSS, inference is based on the max-product belief propagation(MPBP) algorithm that involves local message passing. MPBP is known to converge to the MAP values of the variables on cycle-free graph. In addition to MPBP, UFLP proposed a linear programming(LP) relaxation based message passing algorithm, known as max-product linear programming(MPLP). The solution from MPLP can be augmented with a greedy algorithm that constructs a solution whose cost is at most three times the optimal for metric UFLP instances, where the customer-facility distance measure satisfies the triangle inequality, thus providing a performance guarantee. On a related note, [16] formulated two-view motion segmentation as a facility location problem and solve it as a LP problem by relaxing the original facility location problem.

[11] expands the scope of FL by considering Capacitated Facility Location(CFL). Each facility now has an upper bound on the number of customers it can be assigned to.

The increased complexity in the consistency function now poses a potential combinatorial challenge. [11] shows that tractability can be assured by sorting the messages and consider only the top messages related to the facility capacity. The additional message update due to the global cost function in our work are made tractable and efficient by using similar techniques.

Even though both FLoSS/UFLP and our work are based on AP for solving the motion segmentation problem, there are important differences distinguishing the two works. FLoSS/UFLP solves the classification problem, assuming known motion. Its performance has not been demonstrated on the model selection problem even though, paradoxically, the framework seems to be proposed with this problem in mind. Our proposed work capitalizes on this inherent capability of AP for model selection with the use of a more elaborate facility cost model. Furthermore, our quest for a minimal basis representation drives a more specific subspace hypotheses generation strategy. In FLoSS/UFLP, the subspace hypotheses are generated by random sampling.

[23] analyzed graphical models with high order potentials(HOP), which entails higher order interactions among the discrete variables. A particularly relevant example is the cardinality potential, whose function value is dependent on the number of variables in the subset turned on. The facility cost function we propose in section 4.2 is an instantiation of the cardinality function.

3. Hypothesis generation with minimal basis subspace representation

3.1. Formulation

Our subspace hypotheses generation strategy is based on finding the minimal basis subspace representation for the data matrix. Such parsimonious representation looks for basis common to the overlapping subspaces, thereby reducing the number of basis needed to explain the subspaces. This emphasis on shared basis leads naturally to the joint sparsity formulation (1).

As in SSC and LRR, we use the data matrix itself as the dictionary, and propose the following formulation:

$$\min_{C} \ \|C\|_{2,0} + \gamma \|E\|_{0,2} \quad s.t. \quad \widehat{W}C = \widehat{W} \tag{1}$$

where $\widehat{W} \in \mathbb{R}^{2F \times N}$ is the data matrix constructed from the tracked feature trajectories, $E \in \mathbb{R}^{2F \times N}$ is the column-sparse error matrix, F is the number of frames, N is the number of tracked feature points, $C \in \mathbb{R}^{N \times N}$ is the representation matrix, $\|.\|_{2,0}$ counts the number of non-zero rows and $\|.\|_{0,2}$ counts the number of non-zero columns.

3.2. Convex relaxation

Due to the combinatorial nature and therefore NP-hard nature of (1), we minimize the convex surrogate and model data noise as column sparse outliers, resulting in:

$$\min_{C} \ \|C\|_{2,1} + \gamma \|E\|_{1,2} \quad s.t. \quad \widehat{W}C = \widehat{W} \tag{2}$$

where $\|C\|_{2,1} = \sum_{i=1}^{2F} \sqrt{\sum_{j=1}^{N} ([C]_{ij})^2}$ and $\|E\|_{1,2} = \sum_{j=1}^{N} \sqrt{\sum_{i=1}^{2F} ([E]_{ij})^2}$. (2) is a constrained convex program that can be solved efficiently by the Augmented Lagrange Multiplier(ALM)[21] method. We solve (2) using the Alternating Direction Multiplier Method(ADMM) implementation of the inexact ALM method, as in [18].

Note that our primary motivation for the joint sparsity formulation is to seek the minimal basis representation, whereas in [7], the joint sparsity regularization was introduced to ensure connectivity in the similarity graph generated by encouraging data points from the same subspace to use common representative points from the same subspace. It plays a secondary role so as not to alter the dominance of the ℓ_1 penalty in the objective function.

3.3. Over segmentation

While we have made the sharing of the basis evident(see figure 1), the relaxation artefact(and noise in the data) means that we cannot make use of this result directly to extricate the number of motions and their dependencies. As can be seen from figure 1, the representation matrix contains various artefacts due to the convex relaxation. While the overall two subspace structure is discernible, over segmentation is revealed in the gaps in the rows and the resultant extra rows, making the true number of motion hard to tell. There are in fact 40 subspace hypotheses generated from this convex solution.

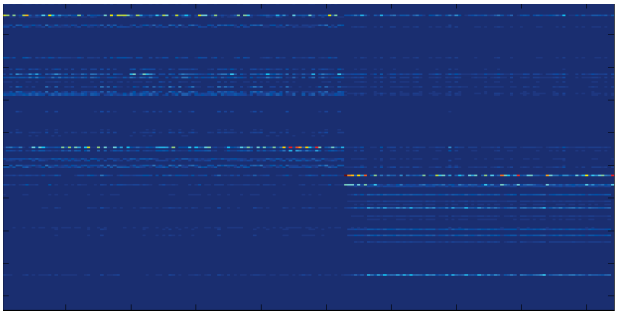

Figure 1. Representation matrix of the truck1 sequence

This over-segmentation phenomenon can be explained by the magnitude dependence of the $\|.\|_{2,1}$ penalty. [22] offers an excellent insight and explanation of this magnitude dependence problem in terms of $\|.\|_1$ in SSC. This magnitude dependence of the convex proxy can be understood

by considering the case when the support has large magnitude. For the counting norm $\|.\|_0$, the magnitude is irrelevant; only the support cardinality matters. For the convex proxy $\|.\|_1$, support entries with large coefficients will result in large $\|.\|_1$ value, imposing an unfair penalty. The magnitude dependence of the $\|.\|_1$ function means that trajectories from the same subspace that are nearly orthogonal will be broken up into two groups, since the large coefficients for self-expression will incur large norm penalty. This explanation also applies for the $\|.\|_{2,1}$ penalty. While some of the numerical methods like reweighted ℓ_1[1] might slightly relief the artefact problems, they do not remove the problems.

Despite the preceding comments, we have now at our disposal much more information. Each column of the coefficient matrix proposes a subspace hypothesis and carries with it a notion of AP responsibility message update to this subspace hypothesis. Row wise, the coefficient matrix indicates the importance of the subspace hypothesis, in terms of the number of trajectory that generates the subspace hypothesis. This is reminiscent of the AP availability message update from the facility. See [8] for more detail about the notion of responsibility and availability. This close relationship lends the joint sparse representation matrix well suited for subspace hypothesis generation.

4. Model selection

Our proposed cost model, which we term as Minimal Basis(MB)-FLoSS, is based on FloSS[15] but with important extensions. These extensions are the facility cost model outlined in section 4.2 which encodes the "ecological" constraint that multiple motions are likely to be dependent, and the discount scheme in section 4.4.2 which ensures that facilities with overlapping basis have lower cost, translating to higher beliefs at these facilities.

Our MB-FloSS method uses the same FLoSS setup and message passing. We thus follow the notations in [9] and [15] in deriving the new message update required by our modified facility cost model.

4.1. FLoSS/UFLP

Due to the relevance of FLoSS/UFLP, we give a quick review here. FLoSS/UFLP formulates the facility location problem in terms of factor graph representation(fig. 2), consisting of variable nodes and factor nodes. This graphical model results in the following objective function:

$$F(\{h_{ij}\}) = \sum_{ij} S_{ij}(h_{ij}) + \sum_{i} I_i(h_{i:}) + \sum_{j} f_j(h_{:j}) \qquad (3)$$

The variables nodes $h_{ij}, i = 1,\ldots,N, j = 1,\ldots,M$, are binary variables that indicate if customer(trajectory) i uses(belongs) to facility(subspace) j, where N is the number of customers and M is the number of facilities. The

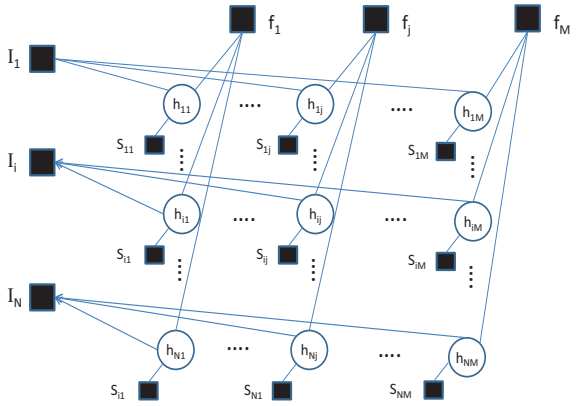

Figure 2. FLoSS factor graph representation

factor nodes evaluate potential functions over the variable nodes they are connected to.

There are three factor potential functions in FLoSS/UFLP. I_i enforces the constraint that one customer chooses one and only one facility. The notation $h_{i:}$ refers to the subset of binary variables connecting customer i to all the facilities from 1 to M. Similarly, the notation $h_{:j}$ refers to the subset of binary variables connecting all the customers from 1 to N to facility j. S_{ij} describes the distance between customer i and facility j. f_j describes the cost when facility j is turned on. Upon convergence of the message update, the binary variables $\{h_{ij}\}$ are turned on if the sum of the messages arriving at the variables are non-negative.

4.1.1 Local facility cost

Due to the key role of facility cost, we describe the FLoSS facility cost model so as to provide a contrast to our proposed cost model. In FLoSS, the subspace hypotheses are generated as random subsets of two, three and four trajectories, thus taking into consideration degenerate subspaces. The cost of a facility is set to be the sum of all pairwise distances between the trajectories forming the subspace. This local cost primarily serves to balance the tendency towards the higher dimensional subspace hypotheses, since higher dimensional subspace hypotheses are able to fit the data better compared to the lower dimensional subspace hypotheses.

Unfortunately, this local cost model does not capture the actual nature of the problem very well, often resulting in the wrong number of facilities being opened. In fact, in FLoSS/UFLP, the number of motion is assumed to be known. Thus they can merge excess number of facilities opened or increase the number of facilities opened by iteratively scaling down the local cost across all facilities.

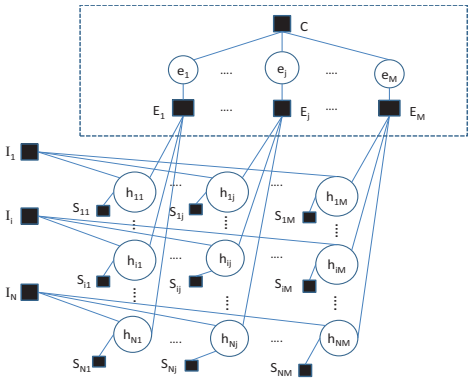

Figure 3. MB-FLoSS factor graph representation. The nodes in the upper rectangular box are extensions to the original FLoSS

4.2. MB-FLoSS facility cost

To address the aforementioned shortcomings, the facility cost function we propose is a global function in the sense that it is a function of the cardinality of the number of facilities opened. Given an upper bound K on the number of motion, we propose a power law facility cost model

$$\mathcal{C} = \begin{cases} ak^p & \text{if } k \text{ facilities are opened, for } k = 1 \text{ to } K \\ \infty & \text{otherwise} \end{cases} \quad (4)$$

where $\mathcal{C}$ is the facility cost function and a, p are constants. Note that $\mathcal{C}$ is a monotonic increasing function of the number of opened facilities. We denote the cost of opening k facilities as $\mathcal{C}_k$. This power law cost model is motivated by the observation that in real life scenes, the larger the number of motions, the more unlikely it is for all of them to be independent. In other words, it reflects not only the cost of increasing complexity with more models, but also the "surprise" of seeing all of them independent from one another. This cost/surprise is only attenuated if there are dependencies between the multiple motions, which will be taken care of by the discount scheme in section 4.4.2.

With the global facility cost function (4), the factor graph representation needs to be modified, as shown in figure 3. The facility cost potential function is now connected to the binary variables $\{e_j\}$. The number of facilities turned on is indicated by the number of $\{e_j\}$ nodes set to 1. The facility cost function $\mathcal{C}$ is therefore a function of $\{e_j\}$. This change will now necessitate message passing involving $\{e_j\}$, reflected in figure 4

4.3. Objective function

The one customer-one facility constraint remains:

$$I_i(h_{i:}) = \begin{cases} 0 & \text{if } \sum_j h_{ij} = 1 \\ -\infty & \text{otherwise} \end{cases} \quad (5)$$

The consistency constraint that ensures that if a customer chooses a facility, the facility gets turned on, also stays:

$$E_j(h_{:j}, e_j) = \begin{cases} 0 & \text{if } e_j = \max_i h_{ij} \\ -\infty & \text{otherwise} \end{cases} \quad (6)$$

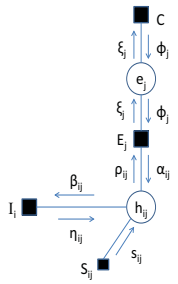

Figure 4. MB-FLoSS factor graph messages

The objective function to be maximized is now

$$F(\{h_{ij}\}, \{e_j\}) = \sum_{ij} S_{ij}(h_{ij}) + \mathcal{C}(\{e_j\}) + \sum_i I_i(h_{i:}) + \sum_j E_j(h_{:j}, e_j) \quad (7)$$

4.4. Message passing

Since we are dealing with binary variables $\{h_{ij}\}$ and $\{e_j\}$, it appears that we need to send two-valued messages between nodes. As pointed out in [11], we only need to propagate the difference between the message values for its two possible settings. When the message passing terminates, the estimated MAP settings for each binary variable is recovered by summing all of its incoming messages. Each binary variable is set to 1 if the sum of all the incoming messages is non-negative, and 0 otherwise.

The message passing not involving $\{e_j\}$ remains the same as in FLoSS. For more detail on those messages, please refer to [14][9][10][11]. The three new message updates ϕ, ξ and α are explained below.

4.4.1 Message update for ϕ

Recall that we only need to send the difference between the message values corresponding to the two different settings. The message to be sent is then

$$\phi_j = \phi_j(1) - \phi_j(0) \quad (8)$$

where

$$\phi_j(1) = \mu_{\mathcal{C} \to e_j}(1) = \max_{e_k, k \neq j} \left[-\mathcal{C}(e_1, .., e_j = 1, .., e_M) + \sum_{k \neq j} \xi_k(e_k) \right] \quad (9)$$

$$\phi_j(0) = \mu_{\mathcal{C} \to e_j}(0) = \max_{e_k, k \neq j} \left[-\mathcal{C}(e_1, .., e_j = 0, .., e_M) + \sum_{k \neq j} \xi_k(e_k) \right] \quad (10)$$

For (9), since e_j is set as 1, we are looking for the max over one, two , . . . , $K - 1$ other e_j's being turned on. For (10), since e_j is kept fixed as 0, we are then looking for the max over one, two , . . . , K other e_j's being turned on.

Even though (9) and (10) look combinatorial, the messages can be simplified and updated efficiently. Leveraging on the insights offered by [11], we observe that finding the max can be achieved by evaluating the sorted set $\hat{\xi}$ and the associated facility cost over the K upper bound number of facilities, where $\hat{\xi}$ is obtained by sorting $\{\xi_j = \xi_j(1) - \xi_j(0), j = 1, \ldots, M, j \neq k\}$ in descending order. The derivation is included in the supplementary material.

For ease of notation, we introduce the cumulative sum operator:

$$S_{ij} = \sum_{k=i}^{j} \hat{\xi}_k \tag{11}$$

where $\hat{\xi}_k$ is the k^{th} element in the sorted set $\hat{\xi}$. Denote the cost difference

$$\delta_{ij} = \mathcal{C}_i - \mathcal{C}_j \tag{12}$$

For the case of $K = 4$, which is the upper bound used in the experiment section, the message update for ϕ_j in (8) can be shown to be

$$\phi_j = \max \begin{cases} -\max\left[S_{11},\ S_{12} - \delta_{21},\ S_{13} - \delta_{31},\ S_{14} - \delta_{41}\right] \\ -\max\left[\delta_{21},\ S_{22},\ S_{23} - \delta_{32},\ S_{24} - \delta_{42}\right] \\ -\max\left[\delta_{31} - S_{22},\ \delta_{32},\ S_{33},\ S_{34} - \delta_{43}\right] \\ -\max\left[\delta_{41} - S_{23},\ \delta_{42} - S_{33},\ \delta_{43},\ \delta_{43},\ S_{44}\right] \end{cases} \tag{13}$$

The indexing in (13) gives a hint on how the message update can be generalized for the number of motion upper bound K and is included in the supplementary material for further reference.

4.4.2 Facility cost discount scheme

The motivation behind the facility cost discount scheme is to encourage the facilities to have shared basis; the more the number of shared basis, the greater the discount. This discount is applied to the cost (4) so that using this discounted $\mathcal{C}$ used in computing message update in (8) can influence facilities with shared basis to be chosen.

The degree of overlap is based on comparison with a reference subspace set $\mathbb{S}_{ref}$, which contains the set of opened facilities according to the current beliefs. This reference subspace is initialized as facility j whose node $\{e_j\}$ has the largest belief. The belief b_j at node e_j is the sum of all the incoming messages, which is $\xi_j + \phi_j$. The candidate set $\mathbb{S}_{can}$ is initialized to be the remaining members of the entire subspace hypothesis set $\mathbb{S}$.

The idea behind the discount scheme is to iteratively fill $\mathbb{S}_{ref}$ with K subspaces with the largest beliefs, after taking into account the facility cost discount due to overlapping subspace basis. At the i^{th} iteration, the discount is applied to cost $\mathcal{C}_i$. The belief for each subspace in $\mathbb{S}_{can}$ is recomputed with this discounted cost. The subspace with the largest belief will then be removed from $\mathbb{S}_{can}$ and added to $\mathbb{S}_{ref}$. After filling $\mathbb{S}_{ref}$ with K subspace hypotheses, the discounted ϕ values associated with $\mathbb{S}_{ref}$ replace the corresponding ϕ message update computed using (13). This facility cost discount scheme is summarized below:

Algorithm 1(Facility cost discount scheme): Given subspace hypothesis set $\mathbb{S}$, upper bound on the number of motion K, discount factor η

1. Compute the belief at each e_j by summing the incoming messages
2. Initialize the reference subspace S_{ref} as the subspace hypothesis whose e_j has the largest belief
3. Initialize the candidate set $\mathbb{S}_{can}$ as the remaining members in $\mathbb{S}$
4. For $i = 1, \ldots, K$
   5. Compute basis overlap degree d for each subspace $\in \mathbb{S}_{can}$ with the reference subspace S_{ref}
   6. For each subspace $\in \mathbb{S}_{can}$, compute the discounted cost $\mathcal{C}_i' = (1 - \eta d) \times \mathcal{C}_i$ and use this discounted cost to compute ϕ_j based on (13)
   7. Find the subspace with the largest belief. Remove this subspace from $\mathbb{S}_{can}$ and add it to S_{ref}
8. End for

4.4.3 Message update for ξ

The message ξ_j can be interpreted as the overall responsibility to the facility j. For each facility j, let k be the index of the largest element of the set $\{\rho_{ij}, i = 1, \ldots, N\}$. The update can then be shown to be

$$\xi_j = \rho_{kj} + \sum_{i \neq k} \max(0, \rho_{ij}) \tag{14}$$

4.4.4 Message update for α

The other message update that is affected by the global facility function is α. The message update for α can be shown to be

$$\alpha_{ij} = \min[0, \sum_{i \neq k} \max(0, \rho_{ij}) + \phi_j] \tag{15}$$

4.5. Subspace hypothesis generation and selection

We provide a different subspace hypothesis generation strategy from FLoSS/UFLP. Our strategy is based on the solution to (2), C^*. Each column i of C^* represents the coefficients of other trajectories required to represent this trajectory i. Since each trajectory comes from an affine subspace, it needs at most four other trajectories for representation. We therefore retain only the top four largest absolute value coefficients in each column and form a subspace hypothesis using that column. The number of subspace hypothesis

M is therefore the number of unique subspace hypothesis proposed by all the trajectories.

When the MB-FLoSS message update is completed, subspace hypothesis j is chosen as a representation subspace if the belief $\xi_j + \phi_j$ at facility j is non-negative.

5. Experiments

We evaluate the performance of our proposed method on the Hopkins 155 dataset [24] and the augmented Hopkins 380 dataset. The Hopkins 155 dataset consists of 155 sequences of feature points labeled according to their motion. There are 120 two motion sequences and 35 three motion sequences. The dataset consists of three categories: checkerboard, traffic and articulated.

For the Hopkins 155 dataset, we base on KO and state-of-the-art LRR for comparison. For the augmented Hopkins 380 dataset, the good performance and availability of Matlab code [17] makes LRR the choice for comparison.

Model selection in LRR returns predicted number of motion in the range of $1-4$. For our facility cost model, we therefore set the upper bound K as four. The facility cost model used for the experiments is shown in figure 5, with the power law in (4) specified by $a = 0.35$ and $p = 2.7$. The discount factor η used in the facility cost discount scheme(algorithm 1) is set to 0.05.

Since the number of motion is no longer known a priori, we need to generalize the misclassification rate to take into account the wrong number of motion group given by model selection. In [24], the misclassification rate is given by the label permutation with the lowest misclassification rate. For the generalized misclassification rate, the label permutation process is naturally extended to account for the case when the wrong number of motion group is given by model selection. Any groups (either in the segmentation result or in the ground truth) whose labels are not assigned after the label permutation process contribute to the misclassified elements. This generalized misclassification rate thus penalizes both model selection error and error in classifying the trajectories according to their motion.

We find that using SSC for classification, based on the number of motion given by the MB-FLoSS model selection gives the best overall performance. This combination is compared against the state-of-the-art LRR.

5.1. Augmented Hopkins 380

The need for augmenting the dataset arises from two considerations. Firstly, the model selection algorithms should work for arbitrary number of motion. In particular, for the Hopkins 155 dataset, the model selection algorithms should be tested against not just two and three motion but one motion as well. Secondly, the skewed distribution of the number of two vs. three motion sequences distorts the model selection rate, since focusing solely on two motion sequences will lead to good model selection rate. This distortion due to the uneven distribution is illustrated in [3] where [12] shows a better model selection performance by estimating two motion most of the time.

In view of these considerations, we choose to augment the Hopkins 155 dataset with one motion sequences and additional three motion sequences. The one motion sequences are derived from the original two and three motion sequences by treating each motion as a one motion sequence. For example, from the three motion sequence 1R2RC, we derive three sequences of one motion 1R2RC_g1, 1R2RC_g2, 1R2RC_g3. The additional three motion sequences are generated by concatenating the two motion traffic sequences with the foreground one motion sequences derived from the two motion traffic sequences. The summary of this augmented data in table 1 shows a more even distribution in terms of the number of sequence for each number of motion.

No. of motion	One	Two	Three
No. of sequence(original)	0	120	35
No. of sequence(augmented)	135	120	125

Table 1. Summary of the augmented Hopkins 155 dataset

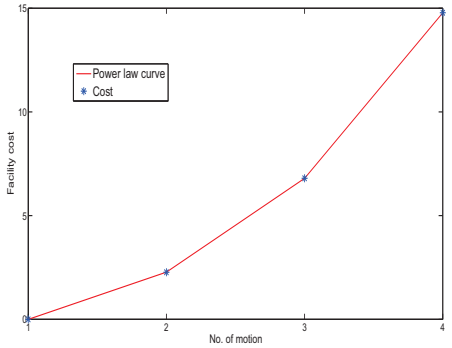

Figure 5. Facility cost model used for the experiments

5.2. Result

Table 2 shows the model selection result for the Hopkins 155 dataset. Our work enjoys an advantage over LRR and outperforms KO decisively. It is worthwhile noting that both LRR and KO show better performance for 2 motion at the expense of 3 motion whereas our proposed method handles both 2 and 3 motion more evenly.

For the augmented Hopkins 380 dataset, table 3 shows the advantage of our proposed work over LRR more decisively. Once again, it is worth noting the more even performance of our proposed work compared to LRR.

The tracked points and basis set chosen for the checkerboard sequence 2rt3rcr_g12 are shown in figure 6 and 7.

For classification, table 4 shows that our proposed method compares favorably to the state-of-the-art LRR.

	MB-FLoSS	LRR	KO
Overall	79.35%(123)	78.06%(121)	74.84%(116)
2 motion	81.67%(98)	84.17%(101)	82.50%(99)
3 motion	71.43%(25)	57.14%(20)	48.57%(17)

Table 2. No. of motion prediction rate for Hopkins 155. The number of sequences predicted correctly is shown in parenthesis

	MB-FLoSS	LRR
Overall	83.68%(318)	81.05%(308)
1 motion	87.41%(118)	85.93%(116)
2 motion	81.67%(98)	84.17%(101)
3 motion	81.60%(102)	72.80%(91)

Table 3. No. of motion prediction rate for the Hopkins 380

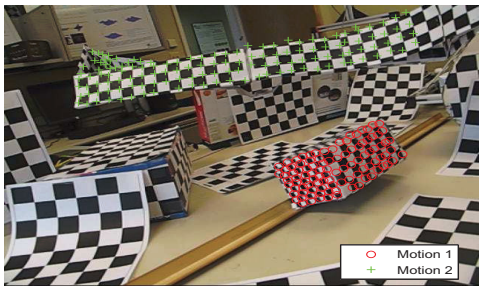

Figure 6. Ground truth for the checkerboard sequence 2rt3rcr_g12

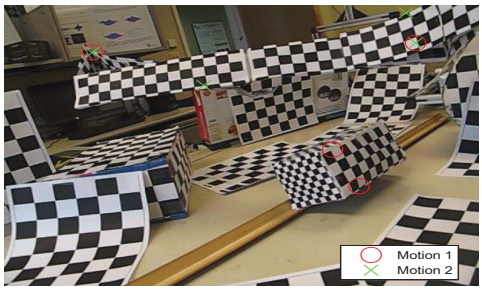

Figure 7. Overlapping basis for the checkerboard sequence 2rt3rcr_g12

	MB-FLoSS + SSC		LRR	
Hopkins	155	380	155	380
Overall	10.04%	8.36%	10.16%	8.98%
1 motion	-	8.74%	-	7.99%
2 motion	9.45%	9.45%	8.59%	8.59%
3 motion	12.07%	6.90%	15.51%	10.43%

Table 4. Generalized misclassification rate for the Hopkins 155 and 380

6. Conclusion

We formulated and realized the minimal basis approach to subspace segmentation and demonstrated its model selection strength. The success hinges on the use of an enhanced FLoSS framework, employing a convex relaxation formulation for subspace hypothesis generation, and a power-law facility cost with a simple discount scheme that favors overlapping subspace. Despite the added complexity due to the modified facility cost, we show how the message passing can be made tractable and efficient.

Acknowledgements. This work was supported by the Singapore PSF grant 1321202075. We like to express our gratitude to Inmar Givoni for her help and guidance.

References

[1] E. J. Candes, M. B. Wakin, and S. P. Boyd. Enhancing sparsity by reweighted l1 minimization. *Journal of Fourier Analysis and Applications*, 14(5-6):877–905, 2008.

[2] T. Chin, D. Suter, and H. Wang. Multi-structure model selection via kernel optimisation. In *CVPR*, pages 3586–3593, 2010.

[3] T. Chin, H. Wang, and D. Suter. The ordered residual kernel for robust motion subspace clustering. In *NIPS*, 2009.

[4] F. Chung. *Spectral graph theory*. Amer Mathematical Society, 1997.

[5] D. Dueck and B. Frey. Non-metric affinity propagation for unsupervised image categorization. In *ICCV*, pages 1–8, 2007.

[6] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *CVPR*, pages 2790–2797, 2009.

[7] E. Elhamifar and R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. *PAMI(Under review)*, 2012.

[8] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315:972–976, 2007.

[9] I. Givoni. *Beyond Affinity Propagation: Message Passing Algorithms for Clustering*. Phd thesis, University of Toronto, 2011.

[10] I. Givoni, C. Chung, and B. Frey. Hierarchical affinity propagation. *arXiv preprint arXiv:1202.3722*, 2012.

[11] I. Givoni and B. Frey. A binary variable model for affinity propagation. *Neural computation*, 21(6):1589–1600, 2009.

[12] K. Kanatani and C. Matsunaga. Estimating the number of independent motions for multibody motion segmentation. pages 7–12, 2002.

[13] F. Lauer and C. Schnorr. Spectral clustering of linear subspaces for motion segmentation. In *ICCV*, pages 678–685, 2009.

[14] N. Lazic, B. Frey, and P. Aarabi. Solving the uncapacitated facility location problem using message passing algorithms. In *AISTATS*, volume 9, pages 429–436, 2010.

[15] N. Lazic, I. Givoni, B. Frey, and P. Aarabi. Floss: Facility location for subspace segmentation. In *ICCV*, pages 825–832, 2009.

[16] H. Li. Two-view motion segmentation from linear programming relaxation. In *CVPR*, pages 1–8, 2007.

[17] G. Liu. Low-rank representation matlab code. https://sites.google.com/site/guangcanliu/.

[18] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *PAMI*, 34(11), 2012.

[19] B. Nadler and M. Galun. Fundamental limitations of spectral clustering. In *NIPS*, volume 19, page 1017, 2007.

[20] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS*, volume 2, pages 849–856, 2002.

[21] J. Nocedal and S. Wright. *Numerical optimization*. Springer, 2006.

[22] M. Soltanolkotabi and E. Candes. A geometric analysis of subspace clustering with outliers. *To appear in Annals of Statistics*, 2011.

[23] D. Tarlow, I. Givoni, and R. Zemel. Hopmap: Efficient message passing with high order potentials. *AISTATS*, 2010.

[24] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *CVPR*, pages 1–8, 2007.

[25] U. Von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.