Split-and-Merge EM for Vine Image Segmentation

Ricardo D. C. Marin
Department of Computer Science
University Of Canterbury
Canterbury, Christchurch
ricardo.castanedamarin@pg.canterbury.ac.nz

Tom Botterill
Department of Computer Science
University Of Canterbury
Canterbury, Christchurch

Richard D. Green
Department of Computer Science
University Of Canterbury
Canterbury, Christchurch

Abstract—**With the goal of recovering the 2D tree structure present on grape vine binary images, in this paper we propose to use Mixture of Gaussians for canes segmentation. The main idea behind our approach is to use information criteria from model selection theory to guide directly the split-and-merge steps for learning Mixture of Gaussians via Expectation Maximization. A novel information criteria we found experimentally is able to adapt to canes at different image scales. We show results of cane segmentation using our criteria in comparison to standard ones as Akaike and Bayesian information criteria. Finally we provide directions on how this work could be extended in the future.**

I. Introduction

In this paper we focus on segmentation of canes of grape vine binary images (see in Figure 4). Such segmentation can be used for estimating and describing the hierarchy of the vine. With this goal, we have developed a method that fit a mixture of Gaussian models to the vine images, learning automatically an appropriate number of components and being flexible at different image scales.

Mixture of Gaussians (MoG) are a powerful tool for clustering and data analysis in image processing and computer vision. In particular, by using a maximum likelihood approach and the Expectation Maximization (EM) algorithm, it is possible to recover the most likely mixture of models that describes some target data. A hard problem in such fitting procedure is the selection of number of components in the mixture, which has to be fixed for running EM. A single component is not enough for most target data, and a component per data point does not provide any new information on the relations between points. The appropriate number of components is located between these two extremes. Now, from a probabilistic perspective the ideal would be to select the number of components that maximizes the likelihood of the data. However, the difficulty arise as the likelihood increase proportional to the number of components, and so the need for regularization methods for learning MoG using a maximum likelihood approach.

The selection of the number of components is not the only issue known for fitting mixtures to data. In [1] Ueda extended the EM algorithm with split-and-merge steps to avoid local maximum convergence. Though the method still requires a fixed max number of models, researchers have used model selection techniques for choosing this value automatically before running the full split-and-merge EM [2], [3]. In turn, model selection is known to successfully rank a set of candidate probabilistic models in order to choose the most appropriate for representing target data. This is done by comparing candidates with respect to a measure of goodness of fit to an underlying ground truth model.

Rather than using model selection for choosing a maximum number of components, in this paper we propose the use of model selection criteria directly for split-and-merge decisions together with an EM algorithm for automatically find the number of components of a MoG. This approach together with a new information criteria we have found experimentally are able to automatically find a scale invariant number of components according to the number of canes present in the vine image.

The rest of this document is structured as follows. Section II reviews the background theory relevant to our method. Section III presents our methods and experimental findings. We show results of applying our new information criteria in comparison with the standard ones in Section IV. Finally Section V presents a discussion and describe ideas that could be used for improving our method in the future.

II. Background

In this section we present previous work and background theory on MoG, EM and model selection. The set of data points will be denoted by $D \subset \mathcal{R}^2$ and will correspond to vine pixel locations of the input binary images.

A. Mixture of Gaussians

Given a parameterized family of Gaussian distributions $\{P(x|\theta_k)\}$ defined for $x \in D$, a K-component MoG is defined as a convex linear combination

$$f_K(x|\Theta) = \sum_{k=1}^{K} \alpha_k P(x|\theta_k)$$

where $\alpha_k \geq 0$ are the mixing weights summing to one, $\Theta = (\theta_1, \alpha_1, \ldots, \theta_K, \alpha_K)$, and each $P(x|\theta_k)$ is called the k-esim component of the mixture. Assuming independence of points in D, the log-likelihood of the mixture is computed as

$$L_{f_K}(D, \Theta) = \sum_{x \in D} \log f_K(x|\Theta). \tag{1}$$

MoG are successfully used for clustering. In this context first we maximize the log-likelihood over D

$$\hat{\Theta} = \underset{\Theta}{\mathrm{argmax}} L_{f_K}(D, \Theta) \tag{2}$$

and then for each $x \in D$ we choose its cluster k_x^* as the one that maximizes the posteriors

$$k_x^* = \operatorname*{argmax}_{k \in \{1,\ldots,K\}} \{\frac{\hat{\alpha}_k P(x|\hat{\theta}_k)}{\sum_{j=1}^{K} \hat{\alpha}_j P(x|\hat{\theta}_j)}\}. \qquad (3)$$

There are some considerations that have to be taken when performing clustering using MoG. In particular we need methods for the optimization in (2) and methods for selecting the number of components K to be used when defining the mixture. We will talk about this issues in more detail in the following subsections.

B. Expectation Maximization

The maximization of the log-likelihood in Eq. (2) is in general a complex task, given the non-convexity inherent in the Likelihood function (1). An standard framework that helps in this process is EM. Some of the main drawbacks of this method are guaranteeing convergence only to a local optimum, and being unstable with respect to initialization. A common practice to overcome these issues is to perform EM several times with random initialization of the parameters. This will generate a family of mixtures $\{f_K^1, \ldots, f_K^l\}$, each mixture with an associated maximum log-likelihood value $\{L_{f_K^1}(D,\hat{\Theta}_1), \ldots, L_{f_K^l}(D,\hat{\Theta}_l)\}$. Finally, the best mixture for the data will be the one that maximizes its likelihood value among the others. We will call this heuristic method EM *with random re-start*.

C. Split-Merge EM

The EM algorithm can be extended to perform clustering with variable number of components. This is done by adding two more steps for splitting and merging components of the mixture, thus increasing and decreasing K automatically. This method was introduced in Ueda [1] as a way to avoid EM convergence to local maximum. The idea is that components that are redundantly describing a part of the data should be merged, while components that under-fit its associated cluster of data should be split.

There exists different criteria for splitting and merging components. The most used split method reported in the literature is based on the *local Kullback divergence* (Local KD) defined to measure the discrepancy between a component and the local data density [1]. If this measure is high with respect to that of the other components, then the component should be split. Similarly, other criteria for splitting components include measures of entropy [3], [14], [15], log-likelihood differences [16], harmony level [19], minimum description length MDL [2], and a test for multivariate normality via the Kolmogorov-Smirnov test [17].

In turn the most used criteria for merging components is a measure of *component correlation*. Ueda [1] observed that when most of the points in a single cluster have posterior probabilities in Equation (3) high and similar for more than one component, then these components should be merged. Other merge criteria include again Local KD [10], [15], Max Entropy

---

For a detailed explanation of EM in the context of computer vision and statistical inference, and implementation details see [4].

[16] and a local analysis on the improvements of a modified Log-Likelihood [18].

There exist multiple ways to incorporate these criteria to the EM framework as well. The standard split-and-merge steps are [8]:

Split Step: Iterate through all components, rank them according to the selected split criteria, and split the one with highest rank.

Merge Step: Compare all pairs of components, rank them according to the selected merge criteria, and merge the pair with highest rank.

These new steps are often added after the EM process has already converged. Denoting by S the split step, by M the merge step, and by PEM a partial EM step corresponding to the use of E and M steps only for the new components; the standard outline of the split-and-merge EM cycle is [8]:

$$EM \rightarrow S \rightarrow PEM \rightarrow EM \rightarrow M \rightarrow PEM$$

Here partial EM steps are added after a split/merge operation, in order for the new components accommodate in respect to the other unaffected components. An important observation here is that a common acceptance test for new models consist on comparing the log-likelihood values of the mixture after and before either split or merge steps [3], [8], [15]. This test allows for another way to stop the whole process, by checking whether there are no more components to split or merge.

To finalize the background section, we will talk about existing methods for identifying an appropriate number of components K on a MoG. Split-and-merge EM approaches usually need a fixed maximum number of components, used for convergence purposes. Though this number can be found using model selection criteria as in [2], [3], we will see in Section III-A that we can use these criteria directly for split-and-merge decisions.

D. Choosing K - Model Selection

Given D fixed, *Model Selection* is the procedure of selecting the *best* model that describes D from a set of candidate models. This selection procedure relies heavily on a measure of goodness-of-fit of a particular model to the data set. In the context of MoG, the set of candidates is a family $\{f_{K_1}, \ldots, f_{K_l}\}$ obtained by varying the number of components K, and model selection refers to finding the best choice of K for the data. In a similar fashion to EM with random re-start, one may think that given this family, the best mixture will be the one that maximizes among the associated maximum likelihoods. However, this may lead to over-fitting one model per sample point, since in general increasing K will increase L_{f_K}. This can be seen as equivalent to making the measure of goodness-of-fit equal to the maximum likelihood value associated to the mixture.

There exists numerous methods for choosing K while avoiding over-fitting [5]–[7]. In this sense, state of the art methods rely on information criteria that estimates the information loss caused by describing the data with a candidate mixture in place of the underlying ground truth model. Some of these criteria are listed below. Here the maximum log-likelihood value will be abbreviated to $L_{f_K} := L_{f_K}(D,\hat{\Theta})$,

and we denote by P_K the number of parameters of a mixture f_K, n is the sample size $|D|$, and AIC, AICc, and BIC stand for Akaike [5], Corrected Akaike [6], and Bayesian [7] information criteria respectively:

$$\text{AIC}_{f_K} = 2(-L_{f_K} + P_K)$$
$$\text{AICc}_{f_K} = \text{AIC}_{f_K} + \frac{2P_K(P_K + 1)}{n - P_K - 1}$$
$$\text{BIC}_{f_K} = -2L_{f_K} + P_K \log n$$

Denoting by IC_{f_K} any of these criteria, model selection among the family $\{f_{K_1}, \ldots, f_{K_l}\}$ is performed by

$$K^* = \underset{K \in \{K_1, \ldots, K_l\}}{\arg\min} \text{IC}_{f_K}$$

and f_{K^*} will be the best fit for the input data.

III. METHODS

We begin by showing that the information criteria seen on Section II-D can be used for split-and-merge operations. Then we analyze their relation to likelihood ratio tests. This will allow us to experiment and further define a novel model selection criteria suitable for segmenting canes on vine images via split-and-merge EM.

A. Split-and-Merge EM Using Information Criteria

Instead of using model selection for finding a maximum number of components as in Section II-D, we can further use it directly as a decision for splitting and merging components. Let $C_k \subset D$ be the cluster of data points associated with the k-esim component. To decide if this component should be split we can construct two mixtures f_1 and f_2 with one and two components respectively, and fit them to C_k using EM. Then, using model selection we can decide how likely this cluster is better represented by one or two components:

$$\text{IC}_{f_2} < \text{IC}_{f_1} \implies \text{SPLIT}$$

where IC could be replaced by any of the model selection criteria seen Section II-D. For merging two components k_1 and k_2, we can repeat the process by using EM to fit f_1 and f_2 to the union cluster $C_{k_1, k_2} = C_{k_1} \cup C_{k_2}$, and then:

$$\text{IC}_{f_1} < \text{IC}_{f_2} \implies \text{MERGE}$$

Observe that the construction and fitting of the two test mixtures f_1 and f_2 can be done using EM with random restart, and their parameters can be used when incorporating the new components to the global mixture; except for the mixture coefficients, since the global mixture may have more than two components. In our experiments, if a component with mixture coefficient α_k is split in two, then the new mixture coefficients can be both initialized to be $0.5\alpha_k$. Similarly we initialized to $\alpha_{k_1} + \alpha_{k_2}$ a new component that resulted from merging components k_1 and k_2. This initialization will ensure that all α_k in the global mixture sum to one.

In this way, to perform a complete Split-Merge EM we can start with an arbitrary number of components and follow the outline of the standard algorithm but now using model selection to guide the process. Note that we do not impose a fixed maximum number of components though this can be incorporated as well if desired. Also note that we don't use any ranking to split or merge components like the methods mentioned in the previous section, but rather our method allows to analyze each component separately. Finally, the convergence of the algorithm can be defined using the convergence of the global log-likelihood values, or where there are no more split or merge operations to be done. Observe that in this way of using model selection, the algorithm will ensure that increases in the likelihood happens locally at all clusters, in contrast to using model selection on two different global MoG.

B. Analysis of Information Criteria for Split-and-Merge

In this section we analyze in general the model selection technique we described in the previous subsection for split-and-merge decisions. Given a set of data points D and two mixtures f_{K_1} and f_{K_2} for which we have computed the maximum likelihood values $L_{f_{K_1}}$ and $L_{f_{K_2}}$ over D, consider the problem of deciding which of the two mixtures fits better the data. Let's further assume that if P_{K_1} and P_{K_2} are the number of parameters of each mixture respectively, then $P_{K_2} = mP_{K_1}$. As described in the previous subsection, we can use model selection over these two mixtures. For instance, if using AIC then we know that f_{K_2} is a better fit than f_{K_1} if

$$\text{AIC}_{f_{K_2}} - \text{AIC}_{f_{K_1}} < 0$$
$$2(-L_{f_{K_2}} + P_{K_2}) - 2(-L_{f_{K_1}} + P_{K_1}) < 0$$
$$L_{f_{K_2}} - L_{f_{K_1}} > (m-1)P_{K_1}$$

Therefore, f_{K_2} is better fit than f_{K_1} if the increase in the likelihood $L_{f_{K_2}}$ is more than the increase in number of parameters from f_{K_1}. Similarly, using the AICc and BIC criteria, we can find

$$\text{AICc}_{f_{K_2}} - \text{AICc}_{f_{K_1}} < 0$$
$$L_{f_{K_2}} - L_{f_{K_1}} > \frac{n(n-1)(m-1)P_{K_1}}{(n-1)^2 - (n-1)(m+1)P_{K_1} + mP_{K_1}^2}$$

and

$$\text{BIC}_{f_{K_2}} - \text{BIC}_{f_{K_1}} < 0$$
$$L_{f_{K_2}} - L_{f_{K_1}} > \frac{(m-1)}{2}P_{K_1} \log n.$$

Thus, in the special case of $P_{K_2} = mP_{K_1}$, these information criteria can be seen as a part of a family of criteria with a threshold function T in the increase in the log-likelihood

$$L_{f_{K_2}} - L_{f_{K_1}} > T(P_{K_1}, n, m) \quad (4)$$

There are some interesting things about this equation. First, the left hand term is proportional to the likelihood ratio, and thus the equation can be seen as a likelihood ratio test parameterized by T. Secondly, if we put $T \equiv 0$ then the goodness-of-fit is going to be exactly the maximum likelihood values L_{f_K}. In particular when $m = 1$, that is, both mixtures have the same number of parameters, T vanishes for all the criteria seen so far. This is exactly the method used in EM with Random re-start, when all mixtures have the same number of parameters and we take the mixture with the highest maximum log-likelihood value. Moreover, the comparison of likelihoods is usually used as an acceptance test for split or merged components in standard split-and-merge EM algorithms [3],
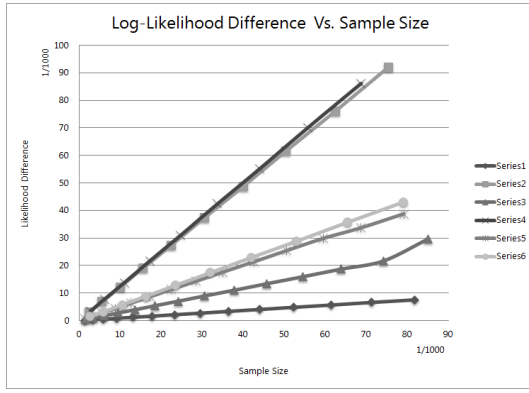
Fig. 1. Maximum log-likelihood difference between mixtures of one and two components, and its dependency on sample size. The six samples sets are based on the images of Figure 3.
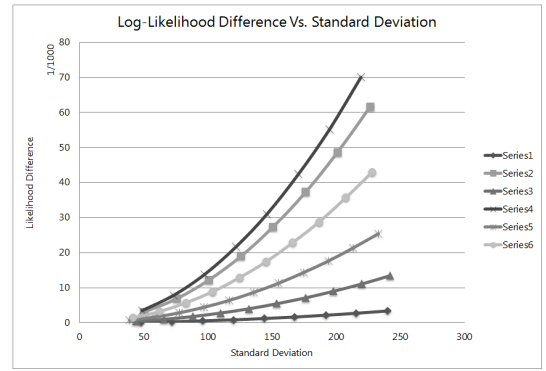

Fig. 2. Maximum log-likelihood difference between mixtures of one and two components, and its dependency on standard deviation. The six samples sets are based on the images of Figure 3.
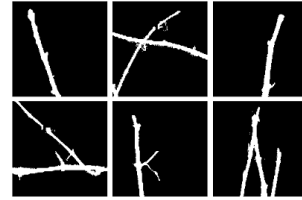

Fig. 3. Test images for experimenting log-likelihood difference dependency on image scale. Extracted from real vine images.

[8], [15]. Also note the function T taken by any of these criteria do not depend on the standard deviation of the data D. However, we have found in our experiments that the difference of the log-likelihood values do depend quadratically an linearly on the standard deviation and sample size respectively. Furthermore we found that by using any of these threshold functions with EM for performing cane segmentation, the results were affected by scale of the image.

Finally, observe that the constraint $P_{K_2} = mP_{K_1}$ is in general met for model selection within a parameterized family of distributions. Here components have always the same number of parameters, but differ in the actual parameter values. Thus the condition is satisfied for deciding between K and mK components. In particular $m = 2$ is exactly the case for the split-and-merge criteria that we described in Section III-A. With the new developments the updated split-and-merge criteria can be written as:

$$L_{f_2} - L_{f_1} > T(P_{K_1}, n, 2) \implies \text{SPLIT}$$
$$L_{f_2} - L_{f_1} < T(P_{K_1}, n, 2) \implies \text{MERGE}$$

This together with a custom threshold function we will define in the next section, will allow us perform split-and-merge EM with automated model selection for segmenting vine canes on binary images and which adapts to image scales.

C. Log-Likelihood Difference Dependency on Image Scale

The main idea behind our method comes from the observation that log-likelihood values in Equation (4) are dependent on some properties of the input data set D. For instance, it is straight forward to see from the likelihood Equation (1) that adding more sample points to D will affect a mixture maximum log-likelihood value. Furthermore, the likelihood depends on the standard deviation of the set D as well. To see this, lets analyze the log-likelihood difference $L_{f_2} - L_{f_1}$ for a set D composed of two cluster of points symmetric around a point O. First consider what happen to L_{f_1} of the mixture with one component, if we move all points in both clusters of D away from O while keeping symmetry. Since the mean parameter will remain unchanged, and the data points will move forward to the Gaussian tails, L_{f_1} will get lower as we move. On the other hand, a mixture with two components will

keep each component centered at each cluster of D even if we move them as before, and thus L_{f_2} will remain constant. As a consequence, the difference of likelihood values $L_{f_2} - L_{f_1}$ as in the left hand of Equation (4) is affected by the standard deviation of D, and it can be made as large as we want.

This fact imply that results of the split-and-merge criteria, as seen in the previous section, will depend strongly on size and sparsity of the sample set. We can see this more clearly in Figures 1 and 2 which shows the maximum log-likelihood difference between mixtures of one and two components and its dependency on sample size and standard deviation of sample sets based on the six images of Figure 3. The images were scaled from original to up 15 scales with and increase of 0.5 from the original size. These plots reveals a linear dependency on sample size and quadratic dependency on standard deviation. Also they allow us to understand how a threshold function as seen in the previous section should adapt to image scales. For instance, consider deciding between a mixture of one or two components for the images with one vine cane. Since at any scale we would want not to split a single component, we see on the graph that our threshold function should greater that these images log-likelihood differences. Similar for vines with two canes or more the threshold function should be less that the log-likelihood differences so that two components will be a better representation for them. This implies that by choosing a threshold function in between the graphs of log-likelihood differences with one and two or more components, the split-and-merge decisions will remain unchanged at any scale. Also note that none of the threshold functions associated to AIC, AICc and BIC satisfy this, and thus they will not generalize well at different image scales.

To construct a threshold function that adapts to image
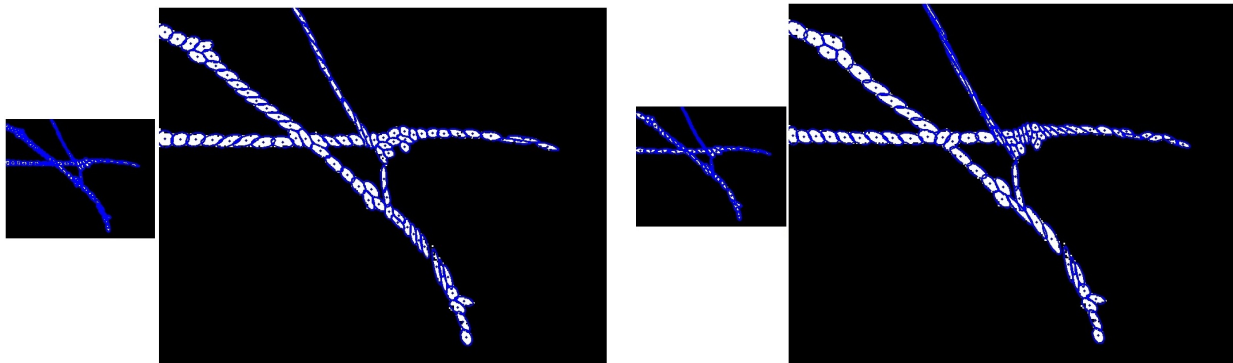
(a) AICc; $T = \frac{5n(n-1)}{(n-1)^2 - 15(n-1) + 50}$. 32 components on both scales.

(b) BIC; $T = 2.5 \log n$. 32 components on both scales.
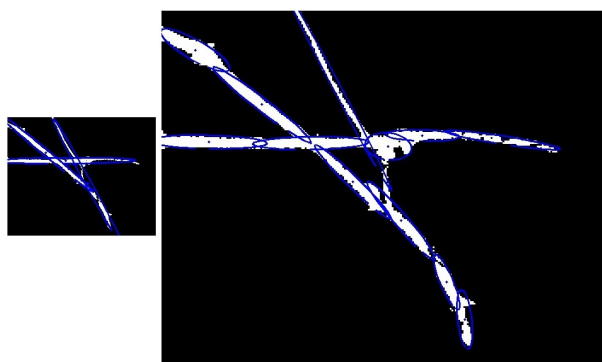
(c) Constant threshold function; $T = 400$. 3 and 14 components respectively.

(d) Our threshold function; $T = 0.1n + 0.01\sigma^2$. 16 and 19 components respectively.

Fig. 4. EM with model selection decisions for split-and-merge. At the top row, AICc (a) and BIC (b) over-fit the data set as a result of its threshold functions being too low and thus splitting almost all components at every iteration. In the bottom row, a constant threshold function (c) perform poorly at a small scale, whereas our threshold function (d) keeps the number of components almost the same at any scale. All procedures where run with a maximum number of components of 32, and with 2 initial components, $m = 2$ and $P_{K_1} = 5$.

scales, we would like to keep the linear and quadratic dependency on sample size and standard deviation respectively:

$$T(n, \sigma) = \frac{1}{2}(An + B\sigma^2) \qquad (5)$$

Observe that we omit the dependency on m and parameter P_{K_1} since they are constant for our split-and-merge decisions with mixtures with one and two components. The coefficients A and B could be estimated using the average values of the slope of the lines of the sample size data, and the average coefficient of the parabolas of the standard deviation data respectively. In particular we used $A = 0.1$ and $B = 0.01$ for the results presented in the next section.

IV. RESULTS

We tested the proposed threshold function on vine images at different scales. Figure 4 shows an example of the results we obtained. We scaled up the images on the left to ten different sizes. All EM iterations involved in the experiments used random re-start for ensuring getting the best convergence parameters. In general split-and-merge decisions based on AIC, AICc and BIC tend to split into several components

over-fitting the data sets. This is due to their low threshold values which allows splitting even for low increases on the log-likelihood. On the other hand, different constant threshold functions work well for particular sample sizes and standard deviations, but don't generalize well when scaling images, yielding too many or too few components depending on whether the image is enlarged or scaled down. In contrast to this, our threshold function adapts to these changes and keeps the number of components equal on average.

V. CONCLUSION

We have presented a split-and-merge EM method that adapts to image scale when learning MoG. We applied the technique for vine canes segmentation on binary images. The method could be extended by further analyzing the threshold function relation to the difference of log-likelihoods of mixtures with one and two components. More in general, note that in many cases the region under consideration for split-and-merge operations is not well represented by neither one nor two components. Therefore we could apply model selection for local clusters with a broader number of components not limited to two, in order to split them into possibly more than

two components. Thus a sensitive analysis of the threshold function with respect to m and number of parameters P_{K_1} could be performed, and could help in generalizing the method to a greater number of components.

Also the learning of the parameters A and B of the proposed threshold function could be studied more in deep. We believe they can be learned from data by using regression methods, or by analyzing directly the difference of likelihoods $L_{diff} = L_{f_2}(D, \hat{\Theta}_2) - L_{f_1}(D, \hat{\Theta}_1)$. We can analyze its dependency with the size and standard deviation of D as $\frac{\partial L_{diff}}{\partial n} = An$ and $\frac{\partial L_{diff}}{\partial \sigma} = B\sigma^2$. L_{diff} is however highly complex as it involves non-linear optimizations for the maximum likelihood estimators in Equation (2).

Finally, we can further research image scale dependency for models other than Gaussians. Though the current method was developed for MoG, general Mixture of Models may present image scale dependency with respect to other properties specific to the probabilistic model used for of each component.

REFERENCES

[1] N. Ueda, R. Nakano, Z. Ghahramani, and G. Hinton, "Split and merge em algorithm for improving gaussian mixture density estimates," in *Neural Networks for Signal Processing VIII, 1998. Proceedings of the 1998 IEEE Signal Processing Society Workshop*, 1998, pp. 274–283.

[2] M. A. T. Figueiredo and A. Jain, "Unsupervised learning of finite mixture models," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 3, pp. 381–396, 2002.

[3] J. MA and T. WANG, "Entropy penalized automated model selection on gaussian mixture," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 18, no. 08, pp. 1501–1512, 2004. [Online]. Available: http://www.worldscientific.com/doi/abs/10.1142/S0218001404003812

[4] S. J. D. Prince, *Computer vision : models, learning, and inference*. Cambridge University Press, 2012. [Online]. Available: http://www.computervisionmodels.com/

[5] H. Akaike, "A new look at the statistical model identification," *IEEE Transactions on Automatic Control*, pp. 716–723, 1973.

[6] T. C.-L. Hurvich, C. M., "Regression and time series model selection in small samples," *Biometrika*, vol. 76, pp. 297–307, 1989.

[7] G. E. Schwarz, "Estimating the dimension of a model," *Annals of Statistics*, vol. 6, pp. 461–464, 1978.

[8] N. Ueda, R. Nakano, Z. Ghahramani, and G. E. Hinton, "Smem algorithm for mixture models," *Neural Comput.*, vol. 12, no. 9, pp. 2109–2128, Sep. 2000. [Online]. Available: http://dx.doi.org/10.1162/089976600300015088

[9] Z. Zhang, "EM algorithms for Gaussian mixtures with split-and-merge operation," *Pattern Recognition*, vol. 36, no. 9, pp. 1973–1983, Sep. 2003. [Online]. Available: http://dx.doi.org/10.1016/s0031-3203(03)00059-1

[10] B. Zhang, C. Zhang, and X. Yi, "Competitive em algorithm for finite mixture models." *Pattern Recognition*, vol. 37, no. 1, pp. 131–144, 2004. [Online]. Available: http://dblp.uni-trier.de/db/journals/pr/pr37.html#ZhangZY04

[11] Y. Li and L. Li, "A split and merge em algorithm for color image segmentation," in *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, vol. 4, 2009, pp. 395–399.

[12] T. Covoes and E. Hruschka, "Splitting and merging gaussian mixture model components: An evolutionary approach," in *Machine Learning and Applications and Workshops (ICMLA), 2011 10th International Conference on*, vol. 1, 2011, pp. 106–111.

[13] Q. Zhao, V. Hautamki, I. Krkkinen, and P. Frnti, "Random swap em algorithm for gaussian mixture models," *Pattern Recognition Letters*, vol. 33, no. 16, pp. 2120 – 2126, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865512002140

[14] G. Celeux and G. Soromenho, "An entropy criterion for assessing the number of clusters in a mixture model," *Journal of Classification*, vol. 13, no. 2, pp. 195–212, 1996. [Online]. Available: http://dx.doi.org/10.1007/BF01246098

[15] K. Blekas and I. E. Lagaris, "Split-merge incremental learning (smile) of mixture models," in *Proceedings of the 17th international conference on Artificial neural networks*, ser. ICANN'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 291–300. [Online]. Available: http://dl.acm.org/citation.cfm?id=1778066.1778100

[16] Y. Li and L. Li, "A novel split and merge em algorithm for gaussian mixture model," in *Natural Computation, 2009. ICNC '09. Fifth International Conference on*, vol. 6, 2009, pp. 479–483.

[17] J. Ma and Q. He, "A dynamic merge-or-split learning algorithm on gaussian mixture for automated model selection," in *Intelligent Data Engineering and Automated Learning - IDEAL 2005*, ser. Lecture Notes in Computer Science, M. Gallagher, J. Hogan, and F. Maire, Eds. Springer Berlin Heidelberg, 2005, vol. 3578, pp. 203–210. [Online]. Available: http://dx.doi.org/10.1007/11508069_27

[18] L. J. Latecki, M. Sobel, and R. Lakaemper, "New em derived from kullback-leibler divergence," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '06. New York, NY, USA: ACM, 2006, pp. 267–276. [Online]. Available: http://doi.acm.org/10.1145/1150402.1150434

[19] L. Li and J. Ma, "A byy split-and-merge em algorithm for gaussian mixture learning," in *Advances in Neural Networks - ISNN 2008*, ser. Lecture Notes in Computer Science, F. Sun, J. Zhang, Y. Tan, J. Cao, and W. Yu, Eds. Springer Berlin Heidelberg, 2008, vol. 5263, pp. 600–609. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-87732-5_67

[20] B. Popovi, M. Janev, D. Pekar, N. Jakovljevi, M. Gnjatovi, M. Seujski, and V. Deli, "A novel split-and-merge algorithm for hierarchical clustering of gaussian mixture models," *Applied Intelligence*, vol. 37, no. 3, pp. 377–389, 2012. [Online]. Available: http://dx.doi.org/10.1007/s10489-011-0333-9