Multiphase Scale Segmentation and a Regularized K-means

Sung Ha Kang, Berta Sandberg and Andy Yip*

Abstract

Typical Mumford-Shah-based image segmentation is driven by the intensity of objects in a given image. We consider image segmentation using scale information in this paper. Using the scale of objects, one can further classify objects in the images from using only the intensity value. The scale of an object is not a local value that the procedure needs to be separated into two steps: multiphase segmentation and scale clustering. Having a reliable multiphase segmentation is essential to the first step and we developed a fast automatic data clustering method for the second step. This new clustering model is an extension from the classical k-means model. It uses the sum-of-squares error for assessing fidelity, and the number of data in each cluster is used as a regularization. The model automatically gives a reasonable number of clusters by a choice of a parameter. We explore various properties of this classification model and present different numerical results.

1 Introduction

Image segmentation is used to partition images, that facilitates the identification of certain objects or features in the images. Image segmentation and active contour are widely studied and various extensions are proposed in different settings such as [13, 30, 39, 45]. Since the publication of the Mumford-Shah model [30] and Chan-Vese's successful level set implementation [5], numerous extensions have been explored and various properties have been studied in variational settings. The Chan-Vese model made it possible to identify objects without even a sharp boundary, and the main driving force is the intensity difference of the objects. This model is extended to incorporate more complicated settings such as multi-channel [4], texture [37], and logic model [36].

We propose to classify objects by the scale in addition to the intensity. Typical Mumford-Shah-based image segmentation is driven by the intensity difference, despite of the fact that the size of the objects can also provide a meaningful classification of the objects. The scale of an object is not a local value, that the scale classification needs to be proceeded by a dependable multiphase segmentation. The objective of multiphase segmentation is to identify more than two phases from the given images, and Vese and Chan [46] first generalized the two-phase case using n number of level sets to identify up to 2^n number of phases. Another way to generalize the level set setting is to represent multiple phases by a single level set by considering

*Kang (corresponding author) is with the School of Mathematics, Georgia Institute of Technology (kang@math.gatech.edu). Sandberg is associated with Adel Research, Inc. and the Department of Mathematics, University of California, Los Angeles (berta.sandberg@adelresearch.com). Yip is with the Department of Mathematics, National University of Singapore (andyyip@nus.edu.sg). This work is partially supported by NSF:DMS-0707184 and the Academic Research Grant R146-000-116-112 from National University of Singapore, Singapore.

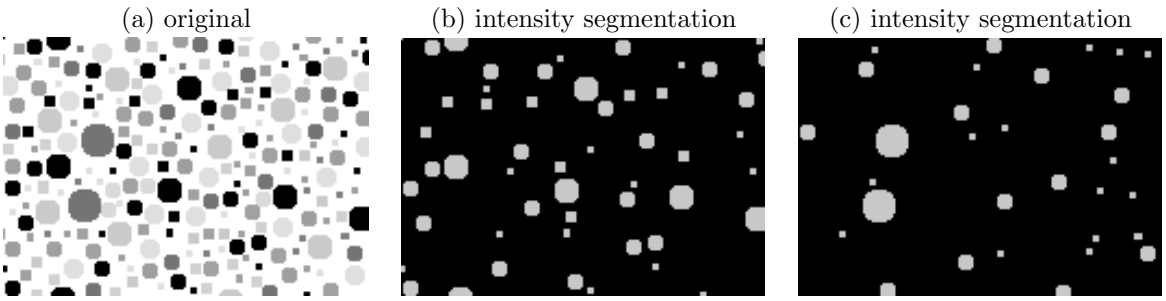

Figure 1: Typical intensity based segmentation. (a) Original given image. (b) and (c) are showing two out of five phases using unsupervised multiphase segmentation model [38]. These are represented using characteristic functions, indicating the location of black and one of darker gray regions. The segmentation is driven by the intensity, that each phase has many different sized objects mixed together.

different layers: Chung and Vese [6] used such approach as well as Lie, Lysaker and Tai [24]. Other related studies include [1, 2, 12, 15, 18, 23, 33, 42].

After intensity based segmentation is performed, we further classify the objects according to the size of each connected component. We propose a new regularized k-means algorithm, which is used to classify the scale values. This model is an extension from the classical k-means model, and explores the connections between data clustering and image segmentation from the modeling point of view. Image segmentation is closely related to data clustering. Their connections are explored by some authors, e.g. a subspace approach and a graph-cut approach for clustering is adapted for image segmentation in [21, 39] respectively.

In this paper, we present an image scale segmentation algorithm using an unsupervised multiphase segmentation model and a new regularized k-means algorithm (Section 2). In subsection 2.1, a fast algorithm for the regularized k-means is proposed for stable realizations of the model, while its stopping criterion gives some insights on how the number of clusters is determined. In subsection 2.2, numerical results illustrate how using scale clustering can improve segmentation results. We further investigate various properties of the regularized k-means model in section 3. Comparisons with the classical k-means are discussed in subsection 3.1, related works of the regularized k-means are listed in 3.2, and numerical experiments showing the performance of the model are presented in subsection 3.3.

2 Image Scale Segmentation

We propose image segmentation using the scale of objects. Using the intensity and the size information together, one can achieve a better classification of objects. Figure 1 shows a motivation of this approach. The original image (a) has many disks with different sizes and intensities. Using intensity based segmentation gives results such as (b) and (c). This is showing two out of five phases, indicating the location of black and one of darker gray regions respectively. The segmentation is driven by the intensity, that each phase has many different sized objects mixed together. For some medical applications, it may be meaningful to cluster the objects according to the size of objects in addition to the intensity.

For the notion of size, we use the scale term

$$\mathcal{S}(A) := \frac{P(A)}{|A|},$$

where $P(A)$ denotes the perimeter of a set A and $|A|$ denotes the 2-dimensional area of a set A. This term is inversely proportional to the size of the object. This notation is used as an automatic parameter scaling in unsupervised multiphase segmentation model [38] and in [41], the inverse is used in the context of total variation (TV) denoising [41]. This term $\mathcal{S}$ is related to the Cheeger Set, where the objective is to find a nonempty set $A \subset \Omega$ of finite perimeter which minimizes $\min_{A\subset\Omega} \mathcal{S}(A)$. This is widely studied in the calculus of variation analysis and some references include [3, 9].

We propose to use scale information for a better classification. Using scale for object classification is considered in a different context and a good literature review on recent approaches can be found in [26]. Unlike intensity based image segmentation, scale is not a local value. Each pixel is not aware of the scale of the connected component it belongs to, until each connected components are identified. Therefore, scale segmentation and intensity based segmentation can not be performed simultaneously and a stable multiphase segmentation is necessary as a pre-precessing step. Any multiphase segmentation method that can separate the objects can be applied. We apply the phase balancing multiphase segmentation model [38] since it is stable and able to handle sensitivity issues of multiphase segmentation, such as initial condition dependence and the pre-assigned number of phases.

With the multiphase segmentation, the image domain Ω is now partitioned by phases, χ_a, according to their intensity value, i.e. $\Omega = \cup_{a=1}^{k} \chi_a$ and $\chi_a \cap \chi_b = \emptyset$ for $a \neq b$. Each phase χ_a may contain many separate connected components, and we further label each connected component by $\chi_{a,b}$ such that

$$\chi_a = \cup_{b=1}^{k_a} \chi_{a,b}, \quad \chi_{a,b} \cap \chi_{a,c} = \emptyset \ \text{ for } \ b \neq c.$$

The scale value for each connected component $\chi_{a,b}$ is computed by $S(\chi_{a,b}) = \dfrac{P(\chi_{a,b})}{|\chi_{a,b}|}$.We define a one-dimensional **data set** to be

$$D = \{S(\chi_{a,b}) | b = 1, 2, \ldots, k_a, \ a = 1, 2, \ldots, k\}.$$

This data set D contains various scale values of objects in the image. The size of D, n is the number of connected components (objects) in the image. We only consider the case when $n < \infty$, that there are finitely many objects which is the case for the discrete setting. For scale segmentation, we cluster this data set D, and classify the scale values and the corresponding objects.

As a data clustering method, we propose the following **regularized k-means** energy,

$$E\left[k, \{I_i\}, \{c_i\} | D\right] = \lambda \left(\sum_{i=1}^{k} \frac{1}{n_i} \right) + \sum_{i=1}^{k} \sum_{d_j \in I_i} |d_j - c_i|^2. \tag{1}$$

Here k is the number of clusters, $n_i = |I_i|$ is the number of data in the cluster I_i, and c_i is the average of data in I_i. In the first term, $\frac{1}{n_i}$ can be generalized to a convex function $f(n_i)$, which satisfies $f(s+t) < f(s)+f(t)$ for all $s, t > 0$. This term $f(n_i)$ is used to introduce the notion of size in the regularization term. The fitting

term measures the spread (intra-cluster dissimilarity) of the clusters, which is the same as the classical k-means.

This model is motivated from the unsupervised multiphase image segmentation [38]. In many applications [6, 18, 24, 42, 46], a number of phases is predetermined or a reasonable estimate is given a priori, and this parallels the case of the classical k-means algorithm. A phase balancing model proposed in [38] addresses this issue of automatically choosing a number of phases k through the minimization of energy functional:

$$E\left[k, \{\chi_i\}, \{c_i\} | u_o\right] = \mu \left(\sum_{i=1}^{k} \frac{P(\chi_i)}{|\chi_i|} \right) \sum_{i=1}^{k} P(\chi_i) + \sum_{i=1}^{k} \int_{\chi_i} |u_o - c_i|^2 dx. \tag{2}$$

Note that in addition to the phases $\{\chi_i\}$ and the intensity averages $\{c_i\}$, the number of phases k is also an unknown variable; only the observed image u_o is given as an input. By minimizing the functional, a reasonable number of phases is found as the image is segmented. The model has one parameter μ that allows different choices for a number of phases, while using μ=1 (for intensity range [0, 255]), unsupervised segmentation is achieved.

The regularized k-means (1) is motivated from the same principle, that we prefer to find a cluster with bigger size compared to having many smaller clusters. Using such regularization term, the model is also able to choose a reasonable number of phases. In the model (1), if λ is set to zero and the number of clusters k is not given, the minimum of the fitting term (the classical k-means) will be achieved when each data point is its own cluster, i.e. $k = n$. The size of the clusters are incorporated into the first term which prefers to have all the data in one cluster. This regularization favors large clusters, and bounds the number of clusters to be $k < n$. Together, the proposed model (1) maintains the tightness of the clusters while avoiding over-fitting by having too many small clusters.

2.1 A regularized k-means algorithm

In variational settings, a typical approach to find a minimizer of the functional is to consider its Euler-Lagrange form and apply a gradient descent method. However, for the proposed regularized k-means model, the number of clusters k is also an unknown that considering a gradient direction requires tedious computation of topological derivatives. Therefore, we propose to directly consider the change in the energy for each data point to find a solution, as in [14, 38, 40]. This is also possible since we are working on a discrete setting of D. The main idea of [40] is to consider the change in the energy directly, that each pixel is moved to the phase which locally minimizes the energy compared to the previous phase. Each iteration of this greedy algorithm is very fast, due to the simple format of the change in the energy functional. As in [38], we also add an option of creating a new cluster to allow changing the number of clusters k.

It is important to notice that this algorithm starts with a simple initial condition: all data are assumed to be in one cluster and the number of clusters is set to be $k = 1$. The model dynamically adjusts the number of clusters during the iteration of algorithm. From the proposed model (1), we consider the energy difference ΔE_{il} for each pair of (i, l), which represents the energy change when a datum $d_j \in D$ is moved from cluster I_i to I_l. For each data $d_j \in D$, the energy change is computed by

$$\Delta E_{il} = E_l - E_i = \lambda \left[\frac{1}{n_i(n_i - 1)} - \frac{1}{n_l(n_l + 1)} \right] + (d_j - c_l)^2 \frac{n_l}{n_l + 1} - (d_j - c_i)^2 \frac{n_i}{n_i - 1}. \tag{3}$$

A regularized k-means algorithm

- Input data set D and λ. Let $k = 1$ and assume all the data are in one cluster.
- Iterate

  Compute the following for each datum d_j for $j = 1, \ldots, n$

  1. For each datum $d_j \in I_i$, compute ΔE_{il} as in (3) for all $l \neq i$, and set $\Delta E_{ii} = 0$. Find the minimum,
  $$value = \min_l \{\Delta E_{il} | l = 1, \ldots, k+1\},$$
  and let $h = \arg\min_l \{\Delta E_{il} | l = 1, \ldots, k+1\}$. Here $k+1$ refers to the new cluster. Then,
  $$\begin{cases} \text{if } value < 0, & \text{move } d_j \text{ to cluster } I_h. \\ \text{if } value \geq 0, & \text{do nothing} \end{cases}$$
  2. Update k if necessary, update $n_i = |I_i|$ and c_i for the clusters that undergo changes.

Table 1: A regularized k-means algorithm

If $\Delta E_{il} > 0$, then the datum d_j will not be moved to I_l since that will increase the energy. If ΔE_{il} is negative, then it is better to move d_j to cluster I_l. A new cluster I_{k+1} with $n_{k+1} = 1$ is created, when the following is the smallest negative value among ΔE_{il} for $l = 1, 2, \ldots, k+1$:

$$\Delta E_{i,k+1} = E_{k+1} - E_i = \lambda \left(\frac{1}{n_i(n_i - 1)} + 1 \right) - (d_j - c_i)^2 \frac{n_i}{n_i - 1}.$$

Table 1 shows the algorithm, where each datum d_j is moved to the cluster that minimizes the energy.

A necessary condition of creating a new cluster can be computed by considering the sign of ΔE. The algorithm will add a new cluster as long as the following holds

$$\lambda \left(1 - \frac{1}{n_i} + \frac{1}{n_i^2} \right) < (d_j - c_i)^2. \tag{4}$$

As n_i increases, $|d_j - c_i|^2$ needs to be bigger to create new clusters, e.g. when one cluster has many data points, it will attract more points, unless it is far from the center c_i. This inequality also shows that λ should be chosen depending on the squared-error $(d_j - c_i)^2$ and n_i (or $f(n_i)$). For example,

$$\lambda \approx C(d_j - c_i)^2 n_i^2 \quad \left(\text{or } C(d_j - c_i)^2 \frac{n_i}{f(n_i)} \right). \tag{5}$$

2.2 Numerical results of scale segmentation

For experiments, after the data set $D = \{S(\chi_{a,b}) | \chi_{a,b} \subset \Omega\}$ is computed, we deleted big scale values (if exist) which represent noise or very small objects. A typical size n of the data set D for image scale segmentation was around $n = 40 \sim 50$ in our experiments.

Figure 2, 3 & 4: general effects of scale segmentation. Figure 2 shows an example where using

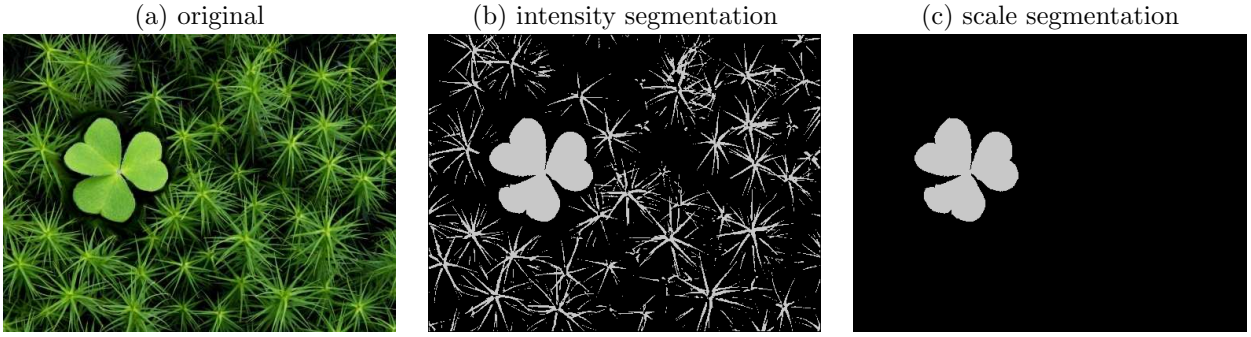

Figure 2: (a) Original given image. (b) The result using intensity based segmentation, showing the light green regions. (c) Further using scale segmentation to identify the clover clearly ($\lambda = 0.25$ is used).

scale segmentation can further improve the segmentation result. Image (b) is using intensity segmentation, where light green regions are all identified together. By further clustering scale values with a regularized k-means algorithm, the clover is clearly identified in image (c). The large clover has different scale value compare to that of the narrow leaves.

Figure 3 is an example using the color blind test image. Since there are some color differences in color-blind test images, using intensity based segmentation method can identify the letters in some of multiple phases. Figure 3 (b) and (c) are showing two phases of intensity based segmentation, where the letter B is clearly identified.

Using the scale segmentation, the objects are classified according to the size of the circles in the images. Third row of Figure 3, image (d), (e) and (f), shows different phases separating objects according to their scale values. Here $\lambda = 0.02$ is used and Figure 3 are showing three out of six phases of scale segmentation. Graph (e) shows the distribution of the scale values in Figure 3 (a). The red intervals indicate the clusters of scale values obtained using the regularized k-means algorithm. We discarded values bigger than 1.5 to disregard noisy small objects. Three phases image (d), (e) and (f) correspond to the first three intervals in the histogram (e), and other three intervals give even smaller objects.

Figure 4 shows a scale segmentation result for a star cluster image. Image (b) and image (c) are two phases (among three phases) using the intensity segmentations. Image (b) represents very bright regions and image (c) less brighter regions. Second row results, image (d) and (e), clearly distinguish different sizes by scale clustering. Bigger clusters are clearly identified in image (d) while smaller stars are in one phase (e).

Figure 5 & 6: medical application, cell images. Original cell image (a) contains many different colors and shape of cells. Using intensity based multiphase image segmentation, the image is separated according to the different intensities: Figure 5 (b) and (c) show two phases (not showing the background phase).

Using a regularized k-means in addition, we can further distinguish the different objects according to their scales. Figure 5 (d) and (e) show two of nine phases separated according to the scale of the objects. The model further separates each connected components in (b) and (c) according to their scale, and makes it easier to identify irregular cells or difference shape objects. Compared to image (b), using scale segmentation

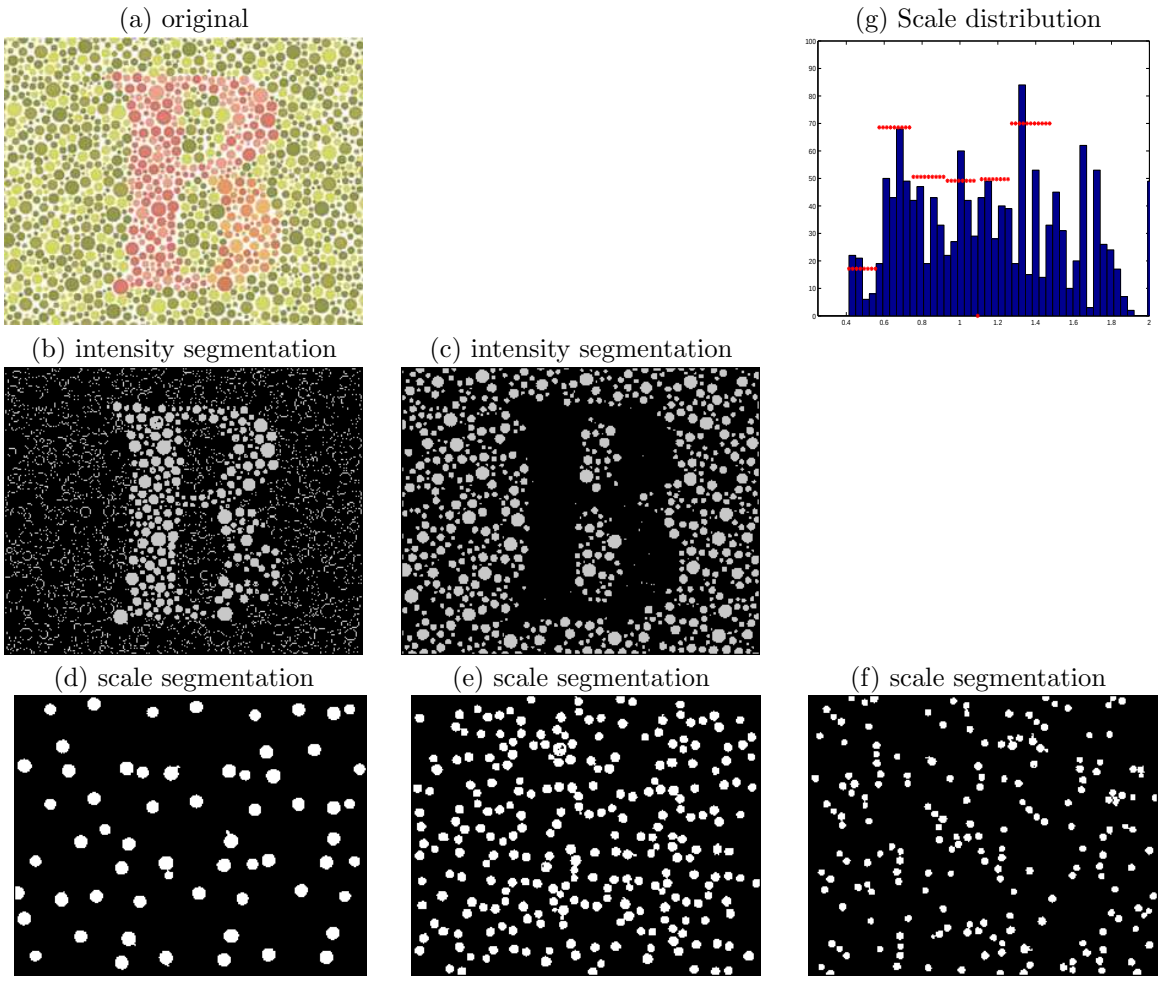

Figure 3: (a) Original given image. (b) and (c) are two phases using intensity segmentation, clearly showing the letter. (d), (e) and (f) are using scale segmentation. Considering only scale of the connected components, the image is separated according to the size of each disks. $\lambda = 0.02$ is used and only showing three phases among six phases. (g) The scale distribution of image (a). We discarded values bigger than 1.5 for more reliable clustering and to remove noise. The red intervals indicate the clustering result of scale data. The three phases (d), (e) and (f) correspond to the first three intervals in the histogram (e).

clearly separates different cells. We clustered scale values below 0.4 and used $\lambda = 0.0005$. The objects with scale value above 0.4 represents small noisy regions and it is shown in Figure 6 (b). Figure 6 (a) shows scale value distribution in histogram form, and the red bars represent each cluster interval using the regularized k-means.

2.3 Some remarks on using scale

The scale term $\frac{P(A)}{|A|}$ only uses boundary length and area. There are some limitations to the kind of objects that it can distinguish. This term will well distinguish disks from thin lines, however, may not be ideal for

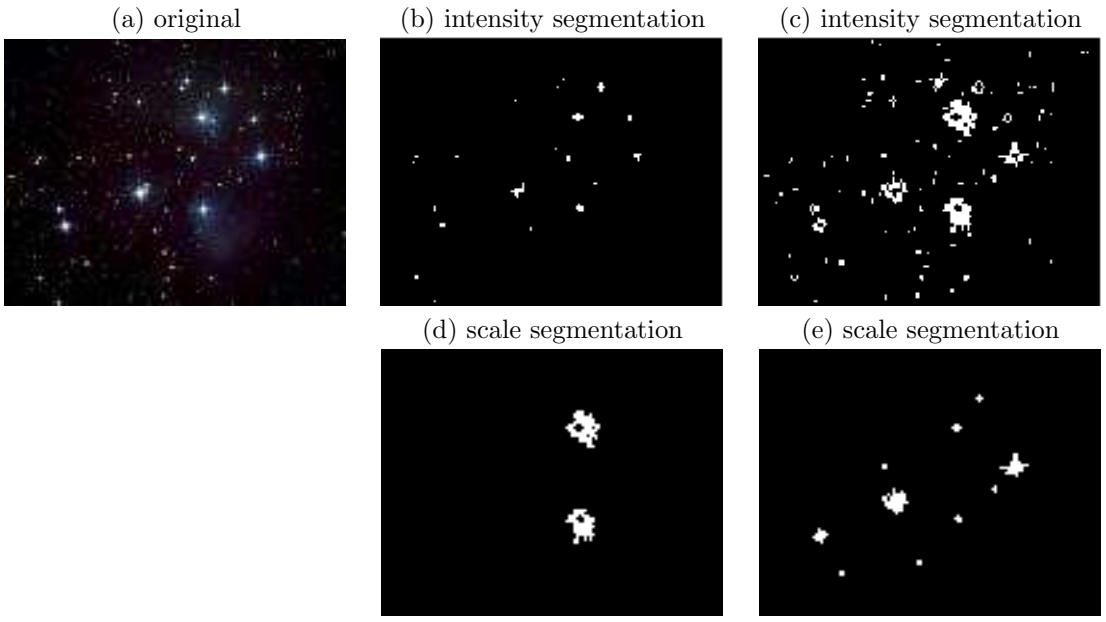

Figure 4: (a) Original given image. (b) and (c) are two phases using intensity segmentation. (d), (e) and (f) are three phases using scale segmentation with $\lambda = 0.5$. Each phase is clearly separated according to the size of the object. Bigger cluster is clearly identified in (d) while smaller stars are in one phase (e).

distinguishing objects with corners. The following property shows an example when different objects can have the same scale value [38].

Property 1 *Any regular (equilateral) polygon B that inscribes a circle with radius r has a scale* $\dfrac{P(B)}{|B|} = \dfrac{2}{r}$. *So does the limiting circle with radius r.*

In addition, since multiphase segmentation is applied first, scale clustering is influenced by the segmentation result. If two same shaped objects are touching and identified as one connected component by the multiphase segmentation, then the two objects will be identified as one bigger object. The scale value of this big object can be different from that of each separate object.

Using scale is different from using a shape prior, but scale segmentation is flexible enough for a general classification. It is not ideal for identifying specific shape, such as finding a particular shaped triangle. Using scale will find triangles and disks of similar scale values, but will distinguish elongated shapes from rounded compact objects.

This approach of scale clustering can be related to histogram segmentation such as [7, 31]. However, a clear difference is that we are proposing a data clustering algorithm, and the histogram clustering result is not directly related to the performance in the object identification.

Just as a note, we have considered some variations of the model to include the scale as well as the intensity in one segmentation model, such as

$$E\left[k, \{\chi_i\}, \{c_i\}, \{s_i\} | u_o\right] = \mu \sum_{i=1}^{k} \left(\frac{P(\chi_i)}{|\chi_i|} - s_i \right) \sum_{i=1}^{k} P(\chi_i) + \sum_{i=1}^{k} \int_{\chi_i} |u_o - c_i|^2 dx,$$

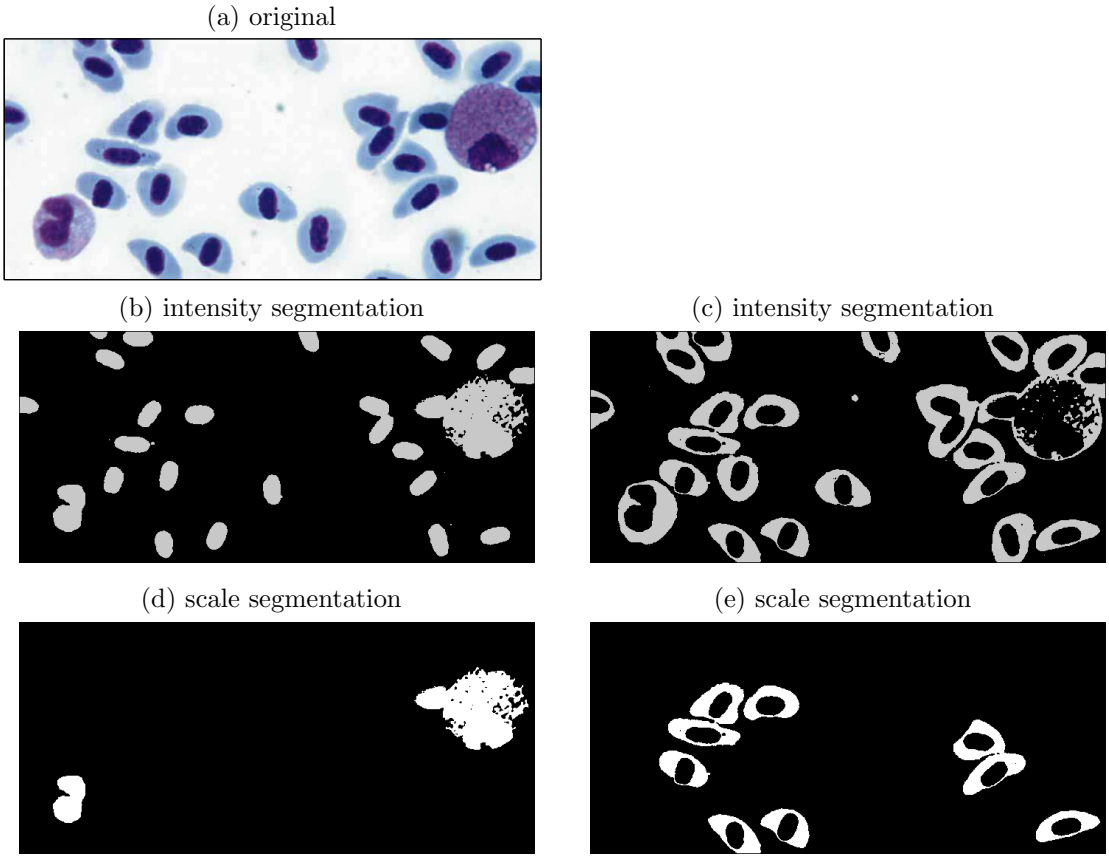

Figure 5: (a) Original given image. (b) and (c) are two phases using intensity segmentation (background phase not shown). Using the regularized k-means, one can further distinguish objects of different scales. (d) and (e) show two of nine phases further separated from image (b) and (c). The cells with different shape/sizes are clearly distinguished compared to only using intensity values.

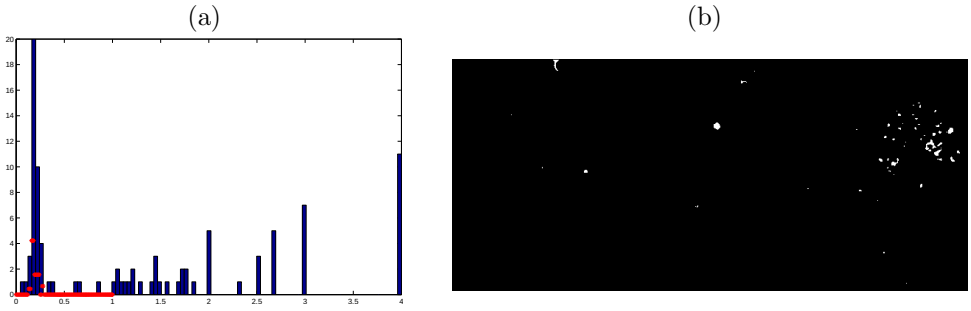

Figure 6: (a) The scale value histogram of Figure 5(a). The red intervals indicate scale clustering using $\lambda = 0.0005$. We discard scale values bigger than 0.4 for more reliable clustering and this value can be roughly chosen to remove the noise. (b) shows all the objects with the scale value above 0.4.

where s_i represents scale values of connected components. This model and variations of such model were not successful, due to a simple but an important fact that the scale is a non-local value. Each pixel is not aware of its own location, that finding the scale value s_i became very unstable. One can consider adding the scale term, however, most likely it will require a two-step algorithm separating scale computation and intensity segmentation.

3 Properties of the Regularized K-means Model

3.1 Comparisons with the classical k-means

For a given data set $D = \{d_1, d_2, \ldots, d_n\} \subset \mathbb{R}^1$, the classical k-means problem formalized by MacQueen [27] is to find a set of $\hat{k}$ clusters which minimizes the following energy,

$$E\left[\{I_i\}|D, \hat{k}\right] = \sum_{i=1}^{\hat{k}} \sum_{d_j \in I_i} |d_j - c_i|^2. \tag{6}$$

Here I_i represents the i-th cluster and c_i is the average over I_i for $i = 1, \ldots, \hat{k}$. The number of clusters $\hat{k}$ is typically given *a priori* or determined by some heuristic steps. When the number of clusters $\hat{k}$ is given, there are many algorithms to compute a solution. Most classical algorithms [11, 16, 25, 27] are based on two basic steps: (1) assign a point to its nearest center; (2) update the cluster centers. These algorithms differ in how the two steps are interlaced. The Ward's method [47] uses an agglomerative hierarchical procedure which locally optimizes the objective for each $\hat{k}$. Some more recent algorithms include deterministic annealing [35], cutting-plane [34], and genetic algorithm [20].

Using the classical k-means model (6), since a fixed number $\hat{k}$ is given, distribution of data will strongly effect the clustering result. In particular, it prefers to have similar distribution (squared error) $|d_j - c_i|^2$ among the $\hat{k}$ clusters. To illustrate, we consider the distribution γ_i in each cluster:

$$\sum_{i=1}^{\hat{k}} \gamma_i^{\min} n_i \leq \sum_{i=1}^{\hat{k}} \sum_{d_j \in I_i} |d_j - c_i|^2 \leq \sum_{i=1}^{\hat{k}} \gamma_i^{\max} n_i.$$

Here $\gamma_i^{\min}$ and $\gamma_i^{\max}$ represent the minimum and maximum distribution within the cluster I_i respectively, and let us assume there exist γ_i^* for $i = 1, 2, \ldots, \hat{k}$ such that

$$\sum_{i=1}^{\hat{k}} \sum_{d_j \in I_i} |d_j - c_i|^2 \approx \sum_{i=1}^{\hat{k}} \gamma_i^* n_i.$$

This setting shows that each cluster will try to have an equal value of $\gamma_i^* n_i$ to minimize the energy among $\hat{k}$ clusters. This is due to minimizing a summation term with a fixed number of clusters $\hat{k}$. To have a similar value of $\gamma_i^* n_i$, a cluster I_1 with many data points will be distributed closer to the center c_1, while another cluster I_2 with a smaller number of data can be distributed farther from the center c_2 (as long as $\gamma_1^* n_1$ and $\gamma_2^* n_2$ are similar). Therefore, the result will be heavily influenced by the choice of k and will strongly depend on how the data are distributed (via $|d_j - c_i|^2$). The proposed model (1) is an improvement over the classical

(a) Regularized k-means (b) classical k-means

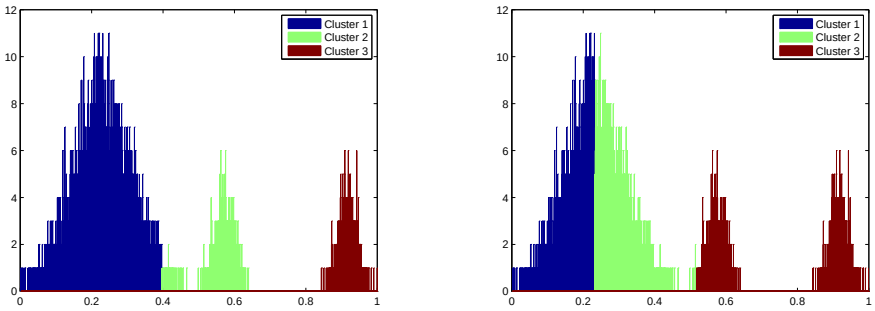

Figure 7: (a) Given data set of $n = 10,000$, and the regularized k-means result using $\lambda = 0.1$. The algorithm gives three clusters. (b) The classical 3-means algorithm.

k-means (6). A similar setting becomes

$$\lambda \sum_{i=1}^{k} \frac{1}{n_i} + \sum_{i=1}^{k} \gamma_i^* n_i.$$

This model has the same fitting term and it will give some balance among the distribution of the clusters. However, the regularization term prefers bigger clusters (a bigger number for each n_i) and stabilizes the process. In addition, k is chosen from the minimizer of the functional, which makes this method flexible. (Flexibility of the choice λ is discussed in Section 3.3.)

3.1.1 Stability of bigger clusters

With a regularization term such as $f(t) = \frac{1}{t}$, the clustering result favors fewer and bigger clusters. Figure 7 is an example showing the stability of the regularized k-means compared to the classical k-means. The size of the data set is $n = 10,000$, among which 70% are drawn from the normal distribution $\mathcal{N}(0.2, 0.06^2)$, 15% from $\mathcal{N}(0.8, 0.02^2)$, and the other 15% from $\mathcal{N}(0.5, 0.02^2)$. The graphs are representing the data in histogram form. Plot (a) is the regularized k-means result using $\lambda = 0.1$ and plot (b) is the classical k-means result with $k = 3$ assigned. The result in (a) is more favorable compared to the result in (b). If the classical k-means algorithm is run multiple times, one can get a similar result to (a). However, it only happened once in about five trials for this example. (The regularized k-means gave the same result regardless of number of trials.) The classical k-means algorithm can be unstable for certain data sets, and the result can be sensitive to the distribution of the data. The proposed model demonstrates more stability.

3.1.2 Stability against outliers

Since the classical k-means algorithm depends strongly on the distribution of data, it can become unstable when outliers or additional datum are introduced. With a regularized k-means this effect is reduced, thanks to flexibility of automatically adjusting the number of clusters. Additional clusters can be created to deal with the outliers, that the clustering of the remaining data is largely unaffected.

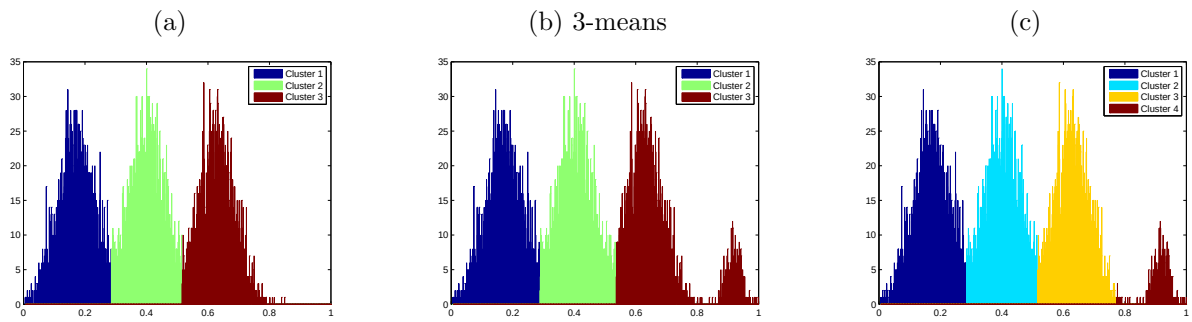

Figure 8: Given data set of $n = 10,000$. (a) using the regularized k-means with $\lambda = 0.1$ and the classical 3-means result (both are exactly the same). (b) A classical 3-means result with new added datum. All intervals are shifted compared to (a) to handle the new datum. (c) using the regularized k-means with $\lambda = 0.1$ and the classical 4-means result (both are the same). Only the third interval is adjusted compared to (a). When the regularized k-means is used, it can handle the data change automatically with a same parameter λ.

Figure 8 (the size of the data set is $n = 10,000$) (a) shows a result of using the regularized k-means with $\lambda = 0.1$, which automatically gave three clusters. This result is exactly the same as using the classical 3-means. When new additional datum are introduced to (a), using the classical 3-means gave (b) as result. Here the intervals are all shifted: in (a) the intervals are I_1=[0.0052,0.2858], I_2=[0,2861, 0.5156], and I_3=[0.5163,0.8490], while in (b), I_1=[0.0052,0.2888], I_2=[0,2892, 0.5368], and I_3=[0.5370,1]. But with the regularized k-means (with the same $\lambda = 0.1$), the algorithm automatically gave 4 clusters as in (c). When the classical 4-means are used, the result was exactly the same as (c).

We can observe a couple of effects: (i) For a clearly separated data set, the classical k-means and the regularized k-means can give exactly the same results, but (ii) when k is not properly changed in the classical k-means, all of the cluster intervals can shift, showing it can easily be effected by outliers. For example, in (a) the intervals are I_1=[0.0052,0.2858], I_2=[0,2861, 0.5156], and I_3=[0.5163,0.8490], and in (c), I_1=[0.0052,0.2858], I_2=[0,2861, 0.5156], I_3=[0.5163,0.7703] and I_4=[0.7723,1]. The first two intervals are not effected, while only the third interval is adjusted to handle the outliers. Notice in (b), all three intervals are shifted to include the new datum.

3.1.3 Effect of the regularization term

The regularization term we used has two basic properties. First, it decreases with k in the sense that if two clusters of size n_1 and n_2 respectively are merged, then the regularization term decreases, i.e. $\frac{1}{n_1+n_2} < \frac{1}{n_1} + \frac{1}{n_2}$. The regularization term is minimized when all data points are assigned to one cluster. Second, clusters of equal sizes are preferred, i.e. $\frac{1}{\frac{1}{2}n} + \frac{1}{\frac{1}{2}n} \leq \frac{1}{n_1} + \frac{1}{n_2}$ for any n_1 and n_2 such that $n_1 + n_2 = n$. In fact, we have

$$\frac{1}{n_1+n_2} < \frac{1}{\frac{1}{2}n} + \frac{1}{\frac{1}{2}n} \leq \frac{1}{n_1} + \frac{1}{n_2}.$$

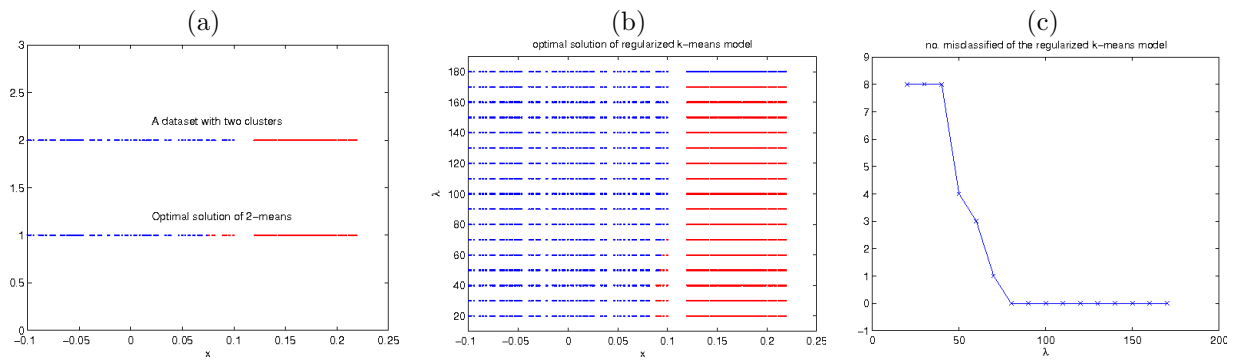

Figure 9: Data contains two clusters of size 100 and 200 respectively. (a) The data set and the optimal 2-means clustering. (b) The optimal solution of the regularized k-means model for $20 \leq \lambda \leq 180$. (c) A plot of the number of misclassified points versus λ.

Therefore, the "small k effect" will eventually overrule the "balancing effect" when the regularization parameter is large enough. Intuitively speaking, decreasing k by one would induce a relatively large increase in the sum-of-squares error. Thus, before λ is large enough to compensate such an increase in the fitting error, the balancing effect takes place to fine tune the clusters (although the changes may not be very significant).

We illustrate these properties in Figure 9. In (a), a data set with two "true" clusters (red and blue) and the optimal 2-means clustering (obtained by enumerating all contiguous clusterings with $k = 2$) are shown. The true blue and red clusters have 100 and 200 points respectively. Due to the difference between the spread of the two clusters, there are 14 points that are "misclassified" by the 2-means method. In (b), the optimal clustering of the regularized k-means with respect to $\lambda = 20, 30, \ldots, 180$ are shown. We remark that when $\lambda \leq 10$, the optimal clustering has $k \geq 3$; when $\lambda \geq 180$, the optimal clustering has $k = 1$. We observe that as λ increases from 20 to 80, the number of misclassified points decreases due to the tendency of the regularization term to balance the size of the clusters. When $80 \leq \lambda \leq 170$, where the size of the two clusters are 100 and 200 respectively, the balancing effect ceases. This is because the left-most points of the red cluster are quite far from the center of the blue cluster. Thus, further balancing would cause a relatively large increase to the sum-of-squares error. When $\lambda \geq 180$, the small k effect takes over so that a single big cluster becomes the optimal choice. The graph (c) depicts the number of misclassified points for $20 \leq \lambda \leq 170$.

3.2 Related work on regularized k-means

Appending a regularization term to the k-means objective has been studied by many authors for different purposes. In [19, 22, 29], an entropy of the cluster membership is added to achieve fuzzy clustering. The fuzziness is controlled through the regularization parameter. The algorithm in [22] is agglomerative which finds a sequence of nested clusterings corresponding to a decreasing sequence of regularization parameters. The final number of clusters is then chosen by using an external cluster validation measure. In [32], an entropy of the cluster sizes is added so that the number of clusters k is part of the objective to optimize. In [44], the number of outliers is added to make the k-means problem less sensitive to outliers.

If we take a broader view that the k-means problem is a special instance of mixture modeling [8, p.526],

then appending a regularization term is equivalent to introducing a prior distribution to the model parameters in the Bayesian framework, under some mild assumptions on the distributions. In particular, several prior distributions of k have been considered. They include the Laplace-empirical criterion, Bayesian inference criterion, minimum description length, minimum message length, Bayesian information criterion, see [28, 10] and the references therein.

In our approach, we do not regularize k directly. Instead, we regularize the size of the clusters via the term $\sum_{i=1}^{k} f(n_i)$ for some convex function f satisfying $f(x+y) < f(x) + f(y)$ for all $x, y > 0$. This favors a smaller k. A similar approach is considered in [32] where an entropy of the cluster sizes is used. We consider those fs that stem from geometric concepts in variational image segmentation. The regularization term in [32] is

$$\sum_{i=1}^{k_{\max}} f(n_i)$$

where $k_{\max}$ is a predetermined upper bound of k and $f(n_i) = -(n_i/n)\ln(n_i/n)$. Here, n_i can take zero value to account for empty clusters. The regularization term used in our model does not allow $n_i = 0$; empty clusters are simply dropped from the expression. This implies that moving one point from a large cluster to a new cluster would induce a relatively large increase in our regularization term (i.e. $f(n_i) \ll f(n_i - 1) + f(1)$) but only a small increase in entropy (i.e. $f(n_i) + f(0) \approx f(n_i - 1) + f(1)$). Thus our model has a great resistance to the formation of small spurious clusters.

Modifying the standard k-means to balance the clusters has been studied in different contexts, e.g. vehicle routing [17] and scheduling in wireless sensor networks [43]. These algorithms are domain specific which take into consideration some special structure of the data. The model proposed in [48] assumes the providence of a set of class labels. The goal is to construct clusters such that each cluster contains a similar number of data from each class and that the cluster size is balanced. Each of the two objectives is achieved by adding a sum-of-squares error term.

3.3 Numerical experiments of the regularize k-means model

Figure 10 (a) shows a given data set of size $n = 20,000$. We use data derived from a histogram of a general image. The parameter λ is changed from 0.1, 0.15, to 0.3, and the algorithm gives four, three and two number of clusters respectively. The given data is reasonably separated among the region with a high peak and other regions. This example shows the effect of the parameter λ.

[λ **vs** n] The stopping criteria (4) and the λ term (5) show that the choice of parameter depends on the size of the data set. We experimented with different sizes of data set, and for a data set of size between $n \approx 100$ and $100,000$, the parameter λ stayed stable. We kept the same distribution while changing the size of the data: 25% of the data are given from the normal distribution $\mathcal{N}(0.3, 0.1^2)$, 25% from $\mathcal{N}(0.6, 0.02^2)$ and the other 50% from $\mathcal{N}(0.8, 0.08^2)$. With the same value of $\lambda = 0.1$, the regularized k-means gives three clusters consistantly. Figure 11 (a) shows one such case of $n = 10,000$. When a significantly small (or large) number of data is used, λ can be adjusted. Figure 11 (b) and (c) show examples when only ten data points are given. With $\lambda = 0.1$ the proposed method gave two clusters (different from three clusters of $n > 100$). With a very small number of data set, it is not clear what is a good clustering result. Using $\lambda = 0.1$, the regularized k-means gave two clusters in (b), and using smaller $\lambda = 0.05$, it gave three clusters in (c).

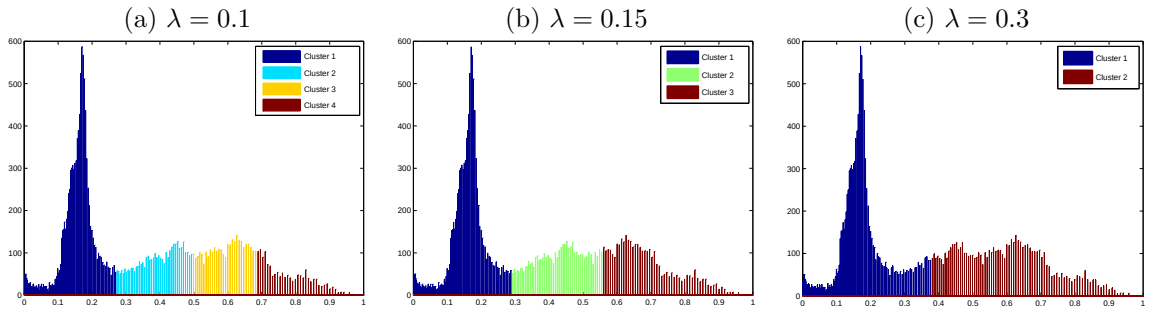

Figure 10: (a) Given data set of $n = 20,000$. (a)-(c) the regularized k-means results using $\lambda = 0.1$, 0.15, and 0.3 and the algorithm gives four, three and two clusters respectively.

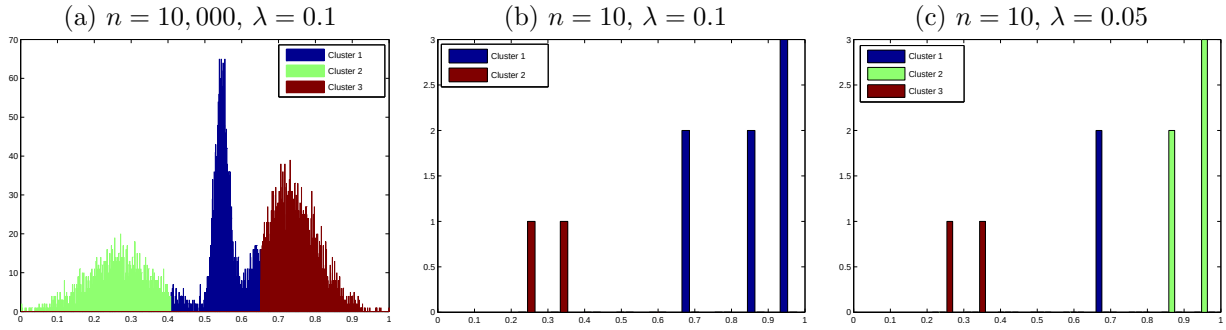

Figure 11: Data is generated with the same distribution while the size of the data set changed from $n = 100,000$ to $n = 100$, and (a) is only showing the case of $n = 10,000$. With $\lambda = 0.1$, all experiments gave three clusters. (b) and (c) are data set of size 10. With $\lambda = 0.1$, the regularized k-means gave two clusters in (b) and using smaller $\lambda = 0.05$, it gave three clusters in (c).

[λ **vs** k] The regularized k-means algorithm does not require any prior knowledge of the number k, however the result can change depending on the choice of λ. It is important to explore how sensitive this choice of λ is compared to the clustering results. Figure 12 shows the graphs of the choice of λ verses the number of clusters k. Clearly the graph is a step function (since k is an integer) and decreasing. Notice there are large flat intervals with the same k values, which shows the stability of the choice of λ. The shape of this graph has some relation to the distribution of the data.

Figure 13 shows the case when only 1% of data is in one cluster. Consider the data generated with $\mathcal{N}(0.3, 0.05^2)$ and $\mathcal{N}(0.7, 0.05^2)$ with different proportions. We fixed the parameter to be $\lambda = 0.1$. Figure 13 is the case when only 1% of data is in the interval around 0.3, but the algorithm finds two intervals. Until 1%, the algorithm seems stable and consistently finds two clusters. Around 1% with $\lambda = 0.1$, sometimes it can also give three clusters, further separating the big cluster into two clusters (b).

[**distribution vs** λ] We tested a data set with two clusters of uniform distribution and normal distribution around $[0.4 - \delta, 0.4 + \delta]$ and $[0.6 - \delta, 0.6 + \delta]$, with a varying δ. The two intervals have the same number of data. For each different δ, λ is set to give two clusters as a result. These experiments show that the parameter λ is dependent on the distribution of the data. As the intervals get wider and closer to each other, a bigger λ is needed to identify the two clusters.

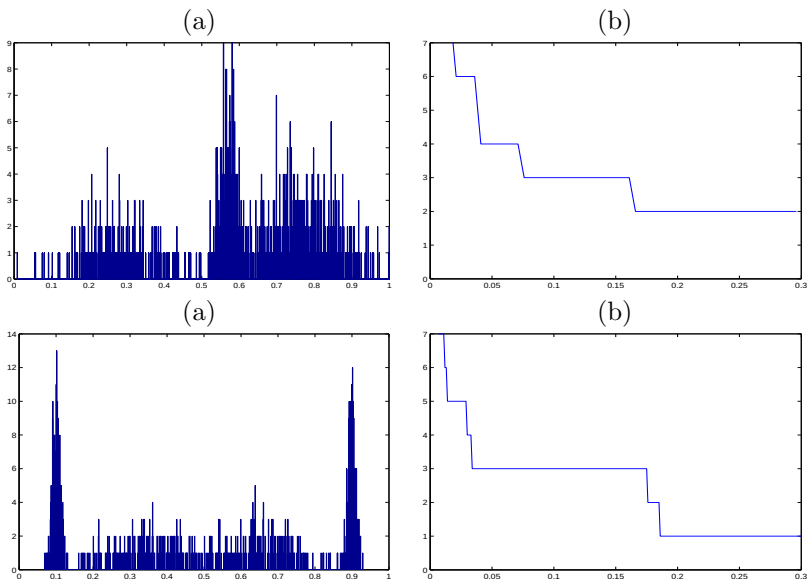

Figure 12: (a) Given data. (b) The graph of λ verses the number of clusters k. Clearly the graph is step function (since k is an integer and decreasing. Notice there are large flat regions with the same k values.

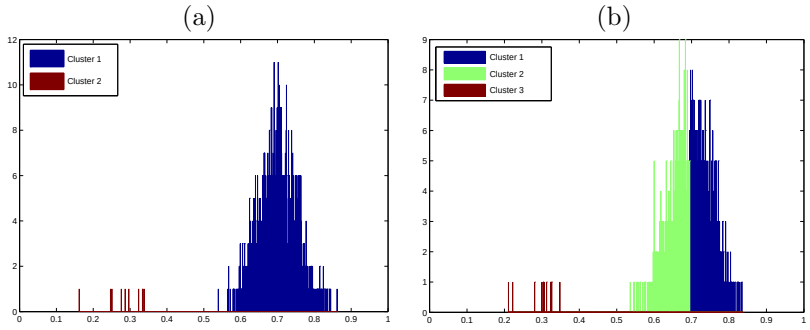

Figure 13: (a) Only 1% of datum are around 0.3, while 99% of the datum are around 0.7. With $\lambda = 0.1$, the algorithm gives two clusters down to 1% around 0.3. At this distribution of 1% and 99%, some trials farther separates the big cluster and give three clusters as in image (b).

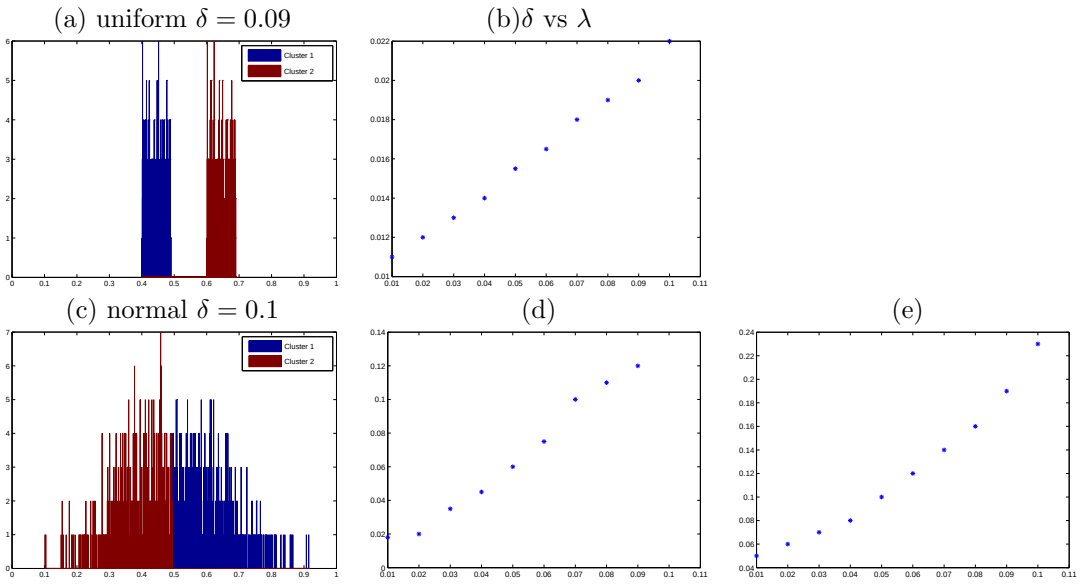

Figure 14: (a) Two clusters of uniform distribution around $[0.4-\delta, 0.4+\delta]$ and $[0.6-\delta, 0.6+\delta]$ with $\delta = 0.09$. (c) Two clusters of normal distribution of $[0.3-\delta, 0.3+\delta]$ and $[0.7-\delta, 0.7+\delta]$ with $\delta = 0.1$. The parameter λ is chosen to result in two clusters .

Figure 14 (a) shows one case of uniform distribution with $\delta = 0.09$, (b) the graph of δ (x-axis) verse λ (y-axis) for uniform distribution. (c) shows the case of normal distribution with $\delta = 0.1$. (d) and (e) are the graph of δ (x-axis) verse λ (y-axis) for normal distribution. The graph (d) has a jump around $\delta \approx 0.06$, since it is where two clusters start to merge and start to look like one cluster as in (c). The graph (e) is the case of two normal distributions at $[0.3-\delta, 0.3+\delta]$ and $[0.7-\delta, 0.7+\delta]$ (data not shown). This case the graph (e) is more uniform compared to (d). Note these values of λ are not a definite value, but one among many possible values, as illustrated in the case of λ vs k in Figure 12. Furthermore, the range of λ varies a little for uniformly distributed data (b), y-axis range [0.01, 0.022], while normal distributed data, the value of λ is larger. The range of the graph in (d) is [0, 0.14] and (e) [0.04, 0.24]. This indicates that for completely separated clusters, the choice of λ can be more sensitive, compared to the case of normal distributed case.

4 Concluding Remark

We explored using scale information for multiphase image segmentation. As shown in the experiments, this allows to further identify the objects more clearly. This application is different from using shape priors, and it clusters the objects according to the scale of objects. Adding the intensity and scale, we can also consider vectorial version of the clustering algorithm.

By exploring image scale segmentation, we propose a data clustering model reduced from variational approach. The model is regularized from the classical k-means using the number of data in each cluster. We consider the function f that stems from geometric concepts in variational image segmentation. We utilized a fast greedy algorithm directly to each data point to efficiently obtain a solution. The proposed regularized k-means algorithm is stable and gives reasonable number of clusters automatically. For example, if the data

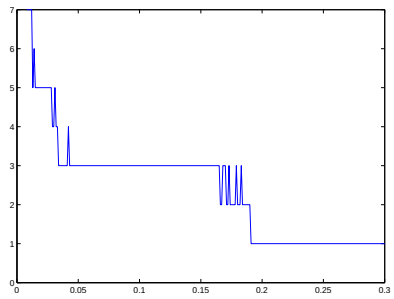

Figure 15: The graph of λ verses the number of cluster k. The experiment is similar to that of Figure 12, except a random reordering of the input datum is used. The graph is a step function and decreasing, but shows some oscillations around the transition. This indicates a possibility of being stuck in the local minimum. However, the result will be reasonable, since the oscillations are only around the transitions and the values are only between two neighboring integers showing the stability of this model.

are scaled between $[0, 1]$, $\lambda = 0.1$ is a reasonable choice. The algorithm is stable with respect to the size of the data $n \approx 100$ to $n \approx 100,000$. The λ and k dependence is also reasonable and gives a stable value of k for a wide range of λ. Various experiments are performed to show the effects of the model.

For this type of NP-hard problem, we are not guaranteed convergence to the global minimum. This algorithm also indicates such a possibility. Figure 15 shows an example similar to Figure 12, but for each different λ, we randomly permuted the order of inputted data D. (The data set itself is the same and only the input order is permuted randomly.) The oscillations around the transitions indicate that the algorithm may get stuck in the local minimum. On the other hand, even if this is the case, the result would be reasonable, since the oscillations are only around the transition between two neighboring intervals where both solutions would make sense. Also, the oscillations are only between two neighboring integers show the stability of this algorithm.

References

[1] E. Bae and X.-C. Tai. Graph cut optimization for the piecewise constant level-set method applied to the multiphase Mumford-Shah model. *the 2nd international conference, SSVM*, pages 1–13, 2009.

[2] T. Brox and J. Weickert. Level set based image segmentation with multiple regions. In *Pattern Recognition*, volume 3175 of *Lecture Notes in Computer Science*, pages 415–423. Springer Berlin / Heidelberg, 2004.

[3] V. Caselles, A. Chambolle, and M. Navaga. Uniqueness of the Cheeger set of a convex body. *Pacific Journal of Mathematics*, 232(1):77–90, 2007.

[4] T. Chan, B. Sandberg, and L. Vese. Active contours without edges for vector-valued images. *Journal of Visual Communication and Image Representation*, 11(2):130–141, 2000.

[5] T. Chan and L. Vese. Active contours without edges. *IEEE Trans. Image Processing* , 10(2):266–277, 2001.

[6] J. Chung and L. Vese. Energy minimization based segmentation and denoising using a multilayer level set approach. *EMMCVPR*, 3457:439–455, 2005.

[7] J. Delon, A. Desolneux, J-L. Lisani, and A-B. Petro. A non parametric approach for histogram segmentation. *IEEE Transactions on Image Processing*, 16(1):253–261, 2007.

[8] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. Wiley-Interscience, New York, 2nd edition, 2001.

[9] A. Figalli, F. Maggi, and A. Pratelli. A note on Cheeger sets. *Proc. Amer. Math. Soc.*, 137:2057–2062, 2009.

[10] M.A.T. Figueiredo and A.K. Jian. Unsupervised learning of finite mixture models. *IEEE Trans. Pattern Anal. Machine Intell.*, 24(3):381–396, March 2002.

[11] E.W. Forgy. Cluster analysis of multivariate data: efficiency vs interpretability of classifications. *Biometrics*, 21:768–769, 1965.

[12] S. Gao and T.D. Bui. Image segmentation and selective smoothing by using Mumford-Shah model. *IEEE Transactions on Image Processing*, 14(10):1537–1549, 2005.

[13] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Trans. Pattern Anal. Machine Intell.*, 6:721–741, 1984.

[14] F. Gibou and R. Fedkiw. Fast hybrid k-means level set algorithm for segmentation. *Proc. of the 4th Annual Hawaii Int. Conf. on Stat. and Math.*, 2002.

[15] T. Goldstein, X. Bresson, and S. Osher. Geometric applications of the split bregman method: Segmentation and surface reconstruction. *Journal of Scientific Computing*, 2009.

[16] J.A. Hartigan and M.A. Wong. A K-means clustering algorithm. *Applied Statistics*, 28:100–108, 1979.

[17] R. He, W. Xu, J. Sun, and B. Zu. Balanced k-means algorithm for partitioning areas in large-scale vehicle routing problem. In *Proc. 3rd Int. Symp. Intelligent Information Technology Application*, volume 3, pages 87–90, Nanchang, China, 2009.

[18] Y.M. Jung, S.H. Kang, and J. Shen. Multiphase image segmentation via Modica-Mortola phase transition. *SIAM Applied Mathematics*, 67:1213–1232, 2007.

[19] N.B. Karayiannis. Meca: maximum entropy clustering algorithm. *IEEE World Congress on Computational Intelligence, Fuzzy Systems*, 1:630–635, 1994.

[20] K. Krishna and M. Narasimha Murty. Genetic k-means algorithm. *IEEE Trans. Systems, Man and Cybernetics—Part B: Cybernetics*, 29(3):433–439, 1999.

[21] Y.N. Law, H.K. Lee, and A.M. Yip. Semi-supervised subspace learning for mumford-shah model based texture segmentation. *Optics Express*, 18(5):4434–4448, 2010.

[22] M.J. Li, M.K. Ng, Y. Cheung, and J.Z. Huang. Agglomerative fuzzy k-means clustering algorithm with selection of number of clusters. *IEEE Transactions on Knowledge and Data Engineering*, 20(11):1519–1534, 2008.

[23] J. Lie, M. Lysaker, and X.-C. Tai. A binary level set model and some applications to Mumford-Shah image segmentation. *IEEE Transactions on Image Processing*, 15(5):1171–1181, 2006.

[24] J. Lie, M. Lysaker, and X.-C. Tai. A variant of the level set method and applications to image segmentation. *AMS Mathematics of Computation*, 75:1155–1174, 2006.

[25] S.P. Lloyd. Least squares quantization in PCM. *IEEE Transaction Information Theory*, 28:129–137, 1982. Technical Note, Bell Laboratories (1957).

[26] B. Luo, J.-F. Aujol, and Y. Gousseau. Local scale measure from the topographic map and application to remote sensing images. *SIAM Journal on Multiscale Modeling and Simulation*, 8(1):1–29, 2009.

[27] J.B. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proc. of 5th Berkeley Symposium on Mathematical Statistics and Probability, Volume I: Statistics*, pages 281–297, 1967.

[28] G. Mchlachlan and D. Peel. *Finite Mixture Models*. John Wiley and Sons, New York, 2000.

[29] S. Miyamoto and M. Mukaidono. Fuzzy c-means as a regularization and maximum entropy approach. In *Proc. Seventh Int'l Fuzzy Sysmtes Assoc. World Congress (IFSA '97)*, volume 2, pages 86–92, 1997.

[30] D. Mumford and J. Shah. Optimal approximation by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.*, 42:577–685, 1989.

[31] K. Ni, X. Bresson, T. Chan, and S. Esedoglu. Local histogram based segmentation using the wasserstein distance. *International Journal of Computer Vision*, 84(1), 2009.

[32] G. Palubinskas, X. Descombes, and F. Kruggel. An unsupervised clustering method using the entropy minimization. *Proceedings. Fourteenth International Conference on Pattern Recognition*, 2:1816 – 1818, 1998.

[33] Y. Pan, D. Birdwell, and S. Djouadi. Bottom-up hierarchical image segmentation using region competition and the Mumford-Shah functional. In *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, pages 117–121, Washington, DC, USA, 2006. IEEE Computer Society.

[34] J. Peng and Y. Xia. A cutting algorithm for the minimum sum-of-squared error clustering. In Hillol Kargupta, Jaideep Srivastava, Chandrika Kamath, and Arnold Goodman, editors, *Proc. of the 5th SIAM Int. Conf. on Data Mining*, pages 150–160, Newport Beach, CA, 2005.

[35] K. Rose, E. Gurewitz, and C.G. Fox. A deterministic annealing approach to clustering. *Pattern Recognition Letters*, 11(9):589–594, 1990.

[36] B. Sandberg and T. Chan. Logic operators for active contours on multi-channel images. *UCLA CAM report 02-12*, 2002.

[37] B. Sandberg, T. Chan, and L. Vese. A level-set and Gabor-based active contour algorithm for segmenting textured images. *UCLA CAM report 02-39*, 2002.

[38] B. Sandberg, S. H. Kang, and T. Chan. Unsupervised multiphase segmentation: A phase balancing model. *IEEE Transaction in Image Processing*, 19:119 – 130, 2010.

[39] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(8):888–905, 2000.

[40] B. Song and T. Chan. A fast algorithm for level set based optimization. *UCLA CAM Report 02-68*, 2002.

[41] D. Strong, J.-F. Aujol, and T. Chan. Scale recognition, regularization parameter selection, and Meyer's G norm in total variation regularization. *SIAM Journal on Multiscale Modeling and Simulation*, 5(1):273–303, 2006.

[42] X.-C. Tai and T. Chan. A survey on multiple level set methods with applications for identifying piecewise constant functions. *International J. Numer. Anal. Modelling*, 1(1):25–48, 2004.

[43] L. Tan, Y. Gong, and G. Chen. A balanced parallel clustering protocol for wireless sensor networks using k-means techniques. In *Proc. 2nd Int. Conf. Sensor Technologies and Applications*, pages 300–305, 2008.

[44] G.C. Tseng. Penalized and weighted k-means for clustering with scattered objects and prior information in high-througput biological data. *Bioinformatics*, 23(17):2247–2255, 2007.

[45] Z.W. Tu and S.C. Zhu. Image segmentation by Data-Driven Markov Chain Monte Carlo. *IEEE Trans. Pattern Anal. Machine Intell.*, 24(5):657–673, 2002.

[46] L. Vese and T. Chan. A multiphase level set framework for image segmentation using the Mumford and Shah model. *International Journal of Computer Vision*, 50(3):271–293, 2002.

[47] J. Ward. Hierarchical grouping to optimize an objective function. *J. Amer. Statist. Assoc.*, 58:236–244, 1963.

[48] S. Yan, H. Zhang, Y. Hu, and B. Zhang. Discriminant analysis on embedded manifold. In Tomáš Pajdla and Jiři Matas, editors, *Proc. 8th ECCV*, pages 121–132, 2004.