

# An Improvement of Color Image Segmentation through Projective Clustering

Song Gao, Chengcui Zhang, and Wei-Bang Chen  
Department of Computer and Information Sciences  
The University of Alabama at Birmingham  
Birmingham, AL 35294, USA  
{gao, zhang, wbc0522}@cis.uab.edu

## Abstract

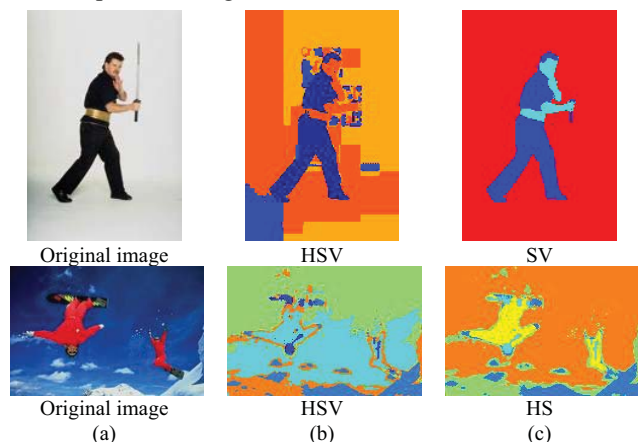
Image segmentation as the processing of partitioning a digital image into multiple segments has wide applications, such as image retrieval, medical inspection, and computer forensics. Clustering methods as one solution are applied on a single or multiple feature spaces of an image, such as color, intensity, or texture, in order to group similar pixels that share certain visual characteristics. Given a particular color image, not all features from a color space, such as RGB, HSV, or Lab, are equally effective in describing the visual characteristics of segments. In this paper, we propose a projective clustering algorithm HCPC (Hill-Climbing based Projective Clustering) which utilizes EPCH (an efficient projective clustering technique by histogram construction) as the main framework and hill-climbing algorithm for dense region detection, for color image segmentation, thereby finding interesting clusters (segments) within subspaces of a given feature space. A new feature space, named HSVrVgVb, is also explored which is derived from HSV (Hue, Saturation, and Value) color space. The experimental results show that compared with hill-climbing algorithm (for efficient color-based image segmentation), our proposed algorithm is more scalable when the dimensionality of feature space is high, and also generates comparable segmentation results.

**Keywords:** Color-based image segmentation, Projective clustering, Hill-climbing algorithm

## 1. Introduction

Image segmentation can be understood as simplifying and/or changing the representation of an image into something that is more meaningful and easier to analyze [1]. For example, in medical applications, digital images of histological slides are used to classify skin biopsies as either melanoma or nevi [2]. In that work, image segmentation as the first step is applied to recognize different tissues in the slides. Of particular interest is the application of content-based image retrieval (CBIR). Color image segmentation especially the segmentation of natural scenes, is one of the fundamental research topics. The segmentation result consists of a set of segments that collectively cover the entire image. Pixels within a segment are similar with respect to certain features, such as color, intensity, or texture, while pixels from different

segments have significant difference in the same feature space. The above property of pixels of segments is consistent with the property of data objects of clusters for a given dataset. Therefore, different clustering algorithms are explored by researchers on different feature spaces in order to find an efficient and/or effective way for image segmentation. Hill-climbing algorithm [3] is one of such algorithms for color-based image segmentation. It detects the local maxima of clusters in the global three-dimensional color (e.g. HSV color space) histogram of an image. Then, the algorithm associates the pixels of an image with the detected local maxima by using k-means clustering with the local maxima as the initial seeds; as a result, several visually coherent segments are generated. The advantage of this algorithm is that it is conceptually simple and relatively fast in low-dimensional spaces. Moreover, it can determine the number of clusters (segments) without any a priori knowledge. However, some meaningful regions of an image may exist in the subspace, but not well represented in the full feature space as exemplified in Figure 1.



**Figure 1.** Segmentation results of the hill-climbing algorithm on different feature sets. (a) The original image, (b) results on HSV color space, (c) results on some 2D subspaces.

Irrelevant or redundant features usually have negative effect on clustering. Since hill-climbing is a global method, which means it finds interesting patterns in the full feature space, it cannot find clusters that are best represented in subspaces. EPCH [4] uses equi-sized histograms to generate “signatures”, where a signature corresponds to a set of dense regions in a set of subspaces, and signatures with regions covering a large enough

number of data objects are identified as subspace clusters. Compared to previous projective clustering algorithms, less prior knowledge on the dataset is needed with EPCH. For example, the dimensionality of the subspaces and the number of natural clusters are no longer required. REVBH (Relative Entropy on Variable Bin-width Histogram) [5] further improves the clustering quality by using variable bin width histograms to describe the local data distribution instead of using equi-sized histograms, and by using relative entropy as a measure to detect dense regions in each subspace. Both algorithms are scalable to dimensionality of dataset. In this paper, we propose a projective clustering algorithm that combines the strength of both EPCH and hill-climbing to explore meaningful regions in subspaces. First, it avoids the usage of global parameters in EPCH. For example, in EPCH, the parameters  $c$  and  $f$ , where  $c < \sqrt{1/f - 1}$  and  $f$  is the proportion of the high density regions in the histogram projection, are applied on each  $k$ -d histogram to detect the dense regions in that  $k$ -d subspace even though the proportion of dense regions in each  $k$ -d space may be different. In the proposed algorithm, hill-climbing finds the local maxima according to the local distribution. Second, the works presented in [4, 5] are both histogram-based projective clustering algorithms. However, a too fine histogram will incur expensive computations while a too rough histogram fails to represent the data distribution. In contrast, hill-climbing algorithm avoids such problem by working directly on the data object level rather than the bin level, finding initial seeds (as the initial cluster centroids for  $k$ -means clustering) in the full dimensional space. However, hill-climbing cannot detect clusters in the subspace, while projective clustering provides such capability. In addition, we also propose a new feature space, named  $HSV_rV_gV_b$ , derived from HSV color space. According to our observation, this new feature space is more suitable for subspace exploration for image segmentation as many segments can be better detected in a subspace of this new space that in the original HSV space.

## 2. $HSV_rV_gV_b$ feature space

Many existing image segmentation algorithms use visual features from multiple modalities such as color, texture, and edge. Their main purpose is to increase the difference between segments by accumulating difference from different modalities. However, fusing information from different modalities is challenging in itself, and the exploration of subspaces in a multimodality space adds another level of difficulty to the already challenging problem. Since the main purpose of this paper is to explore the use of subspace exploration techniques in the domain of image segmentation, we focus on visual features from one single modality only – the color feature space.

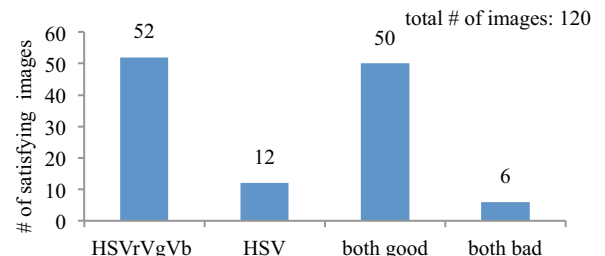
HSV stands for Hue, Saturation and Value. HSV space has been known for its perceptual conformity to how human eye perceives colors. When converting a color image from RGB to HSV space, the  $V$  value of each pixel is calculated as the normalization of the maximum of  $R$ ,  $G$ , and  $B$  values:

$$V = \max(R, G, B)$$

However, some information is lost during this conversion – with one single  $V$  value remaining after conversion, the fact that only one of three color channels contributes to the value of  $V$  is lost. Not only that, it diminishes the difference between two  $V$  values that are equal in values but are contributed by different color channels. Therefore, we divide  $V$  into three independent features, namely  $V_r$ ,  $V_g$ , and  $V_b$ , in order to distinguish from which channel this  $V$  comes from:

$$\begin{aligned} V_r &= \begin{cases} V, & \text{if } R == \max(R, G, B) \\ 0, & \text{otherwise} \end{cases} \\ V_g &= \begin{cases} V, & \text{if } G == \max(R, G, B) \\ 0, & \text{otherwise} \end{cases} \\ V_b &= \begin{cases} V, & \text{if } B == \max(R, G, B) \\ 0, & \text{otherwise} \end{cases} \end{aligned} \quad (1)$$

We then performed hill-climbing algorithm on every subspace of  $HSV_rV_gV_b$  and HSV of 120 color natural scene images, in order to test whether a better segmentation can be obtained from the subspaces of the proposed new feature space compared with HSV subspaces. Figure 2 shows the summary of the comparison result on segmentation quality between  $HSV_rV_gV_b$  and HSV. About half of the images (50 out of 120) yield satisfying segmentation results from the subspaces of both feature spaces, while there are 6 images for which both feature spaces fail to provide a satisfying segmentation result. Among the remaining 64 images,  $HSV_rV_gV_b$  outperforms HSV on 52 images, while HSV produces a better segmentation on the other 12 images. Figure 3 lists some segmentation results for those 120 color images.



**Figure 2.** Comparison of segmentation quality over 120 color images between  $HSV_rV_gV_b$  and HSV feature spaces

In the mean time, we also notice the problem caused by the substitution of three  $V$  channels for one  $V$  channel. One problem is that when the  $V$  value is low in a region, the region is relatively “dark”, and the  $H$  values of those

pixels in that region tend to be randomly distributed over the hue space. Those pixels could be clustered into one group if  $V$  is not split into three channels, while three separate  $V$ s just make the distribution of those pixels in the feature space even sparser. Some unsatisfactory results from certain subspace (or full space) of  $HSV_rV_gV_b$  are shown in Figure 4.



**Figure 3.** (a) The original images, (b) the best segmentation results from certain subspace (or full space) of HSV, (c) the best segmentation results from certain subspace (or full space) of  $HSV_rV_gV_b$

### 3. Projective clustering

As shown in Section 2, satisfying segmentation can often be obtained from subspaces rather than from the full dimensional space of HSV or  $HSV_rV_gV_b$ , therefore, it is necessary to think of a projective or subspace clustering algorithm for image segmentation. Our goal is to detect segments of an image that are best represented in subspaces while also achieving high scalability so that the proposed algorithm scales well up to high dimensional spaces.

According to [6], a generalized projected cluster is defined as: *a set  $\epsilon$  of vectors together with a set  $C$  of data points such that the points in  $C$  are closely clustered in the subspace defined by the vectors  $\epsilon$ . The subspace defined by the vectors in  $\epsilon$  may have much lower dimensionality than the full dimensional space.*



**Figure 4.** (a) The original images, (b) the best segmentation results from certain subspace (or full space) of HSV, (c) best segmentation results from certain subspace (or full space) of  $HSV_rV_gV_b$

#### 3.1 EPCH algorithm

The EPCH algorithm consists of five steps: (1) Constructing  $k$ -d histograms; (2) Detecting dense regions in each histogram; (3) Converting each data object into a signature; (4) Merging similar signature entries to generate cluster candidates; (5) Assigning data objects to corresponding clusters.

Assume that the feature dimensionality of a given dataset is  $D$ . Rather than exploring all possible subspaces in the full dimensional space which is computational very expensive, EPCH chooses to explore at lower-dimensional subspaces with  $k$  dimensions ( $k \leq D$ ) in each of such subspace. The total number of all possible  $k$ -dimensional subspaces is  $C_D^k$ . When a very small  $k$  is considered (e.g.  $k = 1$ ), detected dense regions in such subspace often hide the existence of noise and sparseness in data because of possible overlap of data objects when projected onto lower-dimensional spaces. The noises and/or sparse regions may be exposed in a higher dimensional histogram, such as in a 2d space, but again that is not always true. The higher the dimensionality of histograms, the more expensive the computation is. As a tradeoff, EPCH chooses  $k = 2$  in default.

In the first step, the data range of all dimensions is normalized into  $[0,1]$ . Then, 2-d histograms are constructed for each 2-d subspace. The number of equi-sized bins along each dimension is determined by

Strurges' rule [7], namely  $(1+\log_2 N)$ , where  $N$  is the total number of data objects.

Step 2 is an iterative procedure to detect dense regions in each histogram. The threshold defined in the following equation is driven by the dataset distribution. Dense areas (sets of bins) are removed from current histogram, and the threshold is continuously updated until a termination condition is met.

$$\begin{aligned} \rho &= \mu + \sqrt{1/f - 1} \sigma \\ \text{threshold} &= \rho + c\sigma, c < \sqrt{1/f - 1} \end{aligned} \quad (2)$$

In Equation 2,  $\rho$  represents the density of dense regions under uniform distribution;  $\mu$  and  $\sigma$  are the mean and the standard deviation of the projection distribution;  $f$  is the proportion of high density regions in the histogram projection. Both  $c$  and  $f$  are input parameters as mentioned in Section 1.

In the third step, depending on whether data objects fall into a dense region in each histogram or not, a signature (each entry of a signature represents the ID of that dense region in the corresponding 2-d subspace.) is generated for each data object which includes the IDs of a sequence of dense regions where that data object belongs. The entire signature thus represents the 2-d subspaces corresponding to the dense regions where the data object is projected. According to [4], a signature  $Q_i$  for data object  $O_i$  is an ordered list of  $C_D^k = L$  entries, where the  $j$ -th entry represents the dense region, if any, where the data object is located in subspace  $S_j$ . Specifically,  $Q_i = [Q_{i1}, Q_{i2}, \dots, Q_{iL}]$  where

$$Q_{ij} = \begin{cases} 0, & \text{if the object does not fall into} \\ & \text{any dense region in subspace } S_j, \\ r, & \text{if the object is located in dense} \\ & \text{region } DR_r^{S_j} \text{ in subspace } S_j. \end{cases} \quad (3)$$

Those signatures which represent the same or similar subspaces will be merged to form one signature in Step 4. First, each signature is assigned an initial weight 1. Then, objects with exactly the same signature are merged, and the total number of objects is assigned as the new weight of that signature. After that, all the distinct signatures are sorted in the descending order of their weights. The higher the weight, the more data objects are in the corresponding subspace. The similarity measure is defined in Equation 4. When two signatures are merged to form a new signature, the new weight is the sum of those two signatures' weights. The merging starts by checking the signature with the highest weight in the signature list, if the similarity between that and one of the subsequent signatures is larger than a threshold, the two are merged. The signature with a higher weight serves as the merged signature, and another signature is removed from the list. Then, the next merging iteration starts until no more signatures in the list can be merged. The top *max\_no\_cluster*, which is a pre-specified parameter, of

signatures are returned as the projective descriptions of clusters.

$$\text{similarity} = \frac{\# \text{ of common dense regions}}{\# \text{ of unique dense regions}} \quad (4)$$

In the end, each data object is assigned to a cluster whose signature is the most similar to that of the object's. The similarity is calculated between the signature of a data object and the signature of a cluster. Data objects that do not belong to any cluster (the similarity is below a threshold) are treated as outliers.

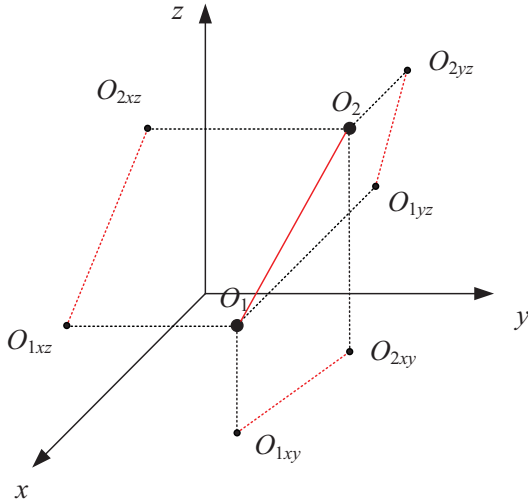
### 3.2 HCPC (Hill-Climbing based Projective Clustering)

REVBH [5] is an improvement of EPCH. It utilizes relative entropy as a density measure to detect dense regions in each  $k$ -d subspace, thereby avoiding the use of global parameters, such as  $c$  and  $f$ , in EPCH. However, REVBH is still a histogram-based projective clustering algorithm. With histogram-based approach, adjacent dense regions which may represent different objects in an image may be detected as one dense region if the bin width is not properly selected. In order to avoid this problem and to reduce the need for a priori knowledge on the dataset, we propose to use the hill-climbing algorithm instead of histogram-based methods for detecting dense regions in the second step, the so-named HCPC (Hill-Climbing based Projective Clustering) algorithm. Initially, a signature of a data object consists of a sequence of IDs of detected clusters in each 2-d subspace where this object belongs to. The advantage of HCPC is that the hill-climbing algorithm can automatically determine the number of clusters existent in each subspace, so there is less a need for the a priori knowledge on the distribution of the dataset in the current subspace. This also eliminates the need for configuring the processes of creating histograms and detecting dense regions. Even though calculating the distance between objects and dense region centers in hill-climbing algorithm is computationally more expensive than that of directly detecting dense bins from histograms in EPCH, the performance does not drop too much when the dimensionality is low (e.g.  $k=2$ ).

Another difference between the proposed algorithm and EPCH is that, in our method of converting each object into a signature, each entry  $Q_{ij}$  in a signature will always have a value greater than zero, while in EPCH a signature entry is either zero (the object does not fall into that subspace) or a positive number representing a region ID (the opposite). This is because the hill-climbing algorithm uses  $k$ -means clustering, with which each data object will "fall into" a dense region in each  $k$ -d subspace whose centroid best matches the data object. Therefore, the value of a signature entry can never be zero. When the hill-climbing algorithm assigns the corresponding dense region ID for each data object in the subspace, the centroid location of each dense region is also recorded. A **centroid vector** is defined for each signature as follows: given a signature  $Q_i = [Q_{i1}, Q_{i2}, \dots, Q_{iL}]$ , its corresponding

center vector is defined as  $C_i = [C_{Q_{i1}}, C_{Q_{i2}}, \dots, C_{Q_{iL}}]$ , where  $C_{Q_{ij}} = (x_{Q_{ij}}, y_{Q_{ij}})$  is a coordinate pair that represent the centroid location of the corresponding dense region in the  $j$ -th 2-d subspace.

There are two phases involved in merging similar object signature entries to generate cluster candidates. First, objects with the same signatures are merged and their weights are summed up to form the new weight for that signature. Then, only those signatures with weight larger than 1 are kept in the list of distinct signatures sorted in descending order of their weight. The second phase is to merge similar signatures. The default similarity threshold in our experiment is 0.75. When one signature has the same similarity with multiple signatures, it will merge with the “closest” one. The maximum 2-d Euclidian distance between two center vectors is used to approximate the real distance between two signatures. As shown in Figure 5, two objects  $O_1$  and  $O_2$  live in a 3-d space, and their projections on the three 2-d subspaces are  $[O_{1xy}, O_{1xz}, O_{1yz}]$  and  $[O_{2xy}, O_{2xz}, O_{2yz}]$ , respectively. The distance between the projections of  $O_1$  and  $O_2$  in each 2-d subspace cannot precisely reflect their real distance in the full dimensional space, but a larger projected subspace distance in general better approximates the real distance than that of a smaller projected distance (e.g.,  $\|O_{1xz}, O_{2xz}\|$  versus  $\|O_{1yz}, O_{2yz}\|$ ). Equation 5 is used to calculate the approximate distance between objects  $O_1$  and  $O_2$ .



**Figure 5.** Distance between two objects in a 3-d space versus their projected distance on each 2-d space

$$dist(Q_1, Q_2) = \max_{1 \leq j \leq L} Eu\_dist(C_{Q_{1j}}, C_{Q_{2j}}) \quad (5)$$

where  $Eu\_dist()$  represents the Euclidian distance. In EPCH, the user is required to specify the expected maximum number of clusters, which is not always available as a priori knowledge. In this work, we try to eliminate the need for this parameter, because it is difficult in image segmentation to specify a fixed number of clusters given a set of images with vastly different

scenes. Instead, we simply put a constraint on the minimum weight of a meaningful signature (e.g.,  $weight \geq 200$  objects/pixels). Recall that those signatures with weight 1 are considered trivial and removed from the distinct signature list prior to the merging of similar signatures. After merging, a post-processing is performed to reassign data objects previously removed with the removal of trivial signatures to one of the remaining signatures that best matches the object and update the signature weights accordingly. Then, signatures with weight smaller than the minimum requirement will be removed from the list. In the final step, each data object is compared against each remaining signature again and is then assigned to the signature with the maximum similarity. Compared with EPCH, one more round of association between data objects and clusters are performed with the proposed algorithm.

When the feature dimensionality of a dataset is very low, the full-dimensional space hill-climbing clustering is faster than HCPC. As the dimensionality increases, the computational cost of hill-climbing increases exponentially, since it detects the local maxima in the full dimensional histograms of the dataset, and the fact that  $k$ -means needs to calculate distance in the full dimensional space makes it even more computationally expensive. Therefore, hill-climbing is not suitable for processing high dimensional data. On the contrary, the time complexity of HCPC increases linearly (with  $k=2$ ) mainly because it does not attempt to detect dense areas directly from the high-dimensional space. Instead, HCPC tries to use a set of dense regions detected from a lower-dimensional space to approximate dense regions in the high-dimensional space. With HCPC, when the dimensionality increases from  $D$  to  $D+1$ , the number of entries in a signature increases by  $C_D^1$ , and the total number of entries in the signature list increases by  $N \cdot C_D^1$ , where  $N$  is the total number of objects.

#### 4. Experiments on HSVrVgVb space

120 color images of natural scenes randomly selected from a 10,000 Corel image dataset (<http://www.cs.princeton.edu/cass/benchmark/>) with dimensions 192x128 or 128x192 are used as the test dataset. All experiments are performed on a PC with Intel(R) Core(TM)2 Duo CPU E8200 2.66GHz, 2GB RAM, and Windows XP SP3. The source code is implemented in MATLAB.

The criterion used to evaluate the segmentation quality during the comparison is defined as follows:

Acceptable Segmentation Quality: If the majority of each salient object in the image can be discovered and represented by a non-trivial segment in the segmentation map, such a result is acceptable. Otherwise, it is not.

We compare the segmentation results of HCPC with that of REVBH and full-dimensional space hill-climbing

algorithm in  $HSV_r V_g V_b$  feature space, respectively, using two performance measures:

- 1) The number of images for which one algorithm outperforms the other on the segmentation quality. If there is no clear winner, the comparison result is thus ‘comparable’. This measure is for the comparison of relative segmentation quality.
- 2) The number of images for which the segmentation results are acceptable. The results produced by the two algorithms in comparison could be both acceptable while one could outperform the other. This is a measure of absolute segmentation quality.

The results in Table 1 show that HCPC can generate more satisfying segmentation results than REVBH according to the above measures. As we mentioned before, histogram-based methods are not sufficient to distinguish adjacent dense regions. In contrast, hill-climbing algorithm in HCPC performs clustering at the pixel level when detecting dense regions in the subspace. Therefore, it is more sensitive in detecting objects in an image.

**Table 1.** Comparison of segmentation results between REVBH and HCPC (total number of images: 120)

Measure	HCPC wins	REVBH wins	Comparable	
1	70	14	36	
Measure 2	HCPC is acceptable but REVBH is not	REVBH is acceptable but HCPC is not	Both acceptable	Both unacceptable
	38	2	78	2

We further compare HCPC with the full-dimensional Hill-Climbing (HC) clustering algorithm on the  $HSV_r V_g V_b$  feature space. As shown in Table 2, HCPC demonstrates a segmentation quality comparable to that of HC algorithm in a relatively low dimensional feature space ( $D=5$ ). However, as we discussed in Section 3, HCPC is much faster than HC when the dimensionality of the feature space grows. In the domain of image segmentation, it is not uncommon to use  $>20$  features during segmentation, especially when features such as textures and edges are considered [8, 9], which could easily make HC computationally infeasible. We will further explore the performance issues of these two algorithms in future work. Figure 6 illustrates some segmentation results produced by these three algorithms.

**Table 2.** Comparison of segmentation results between Hill-Climbing (HC) and HCPC (total number of images: 120)

Measure	HCPC wins	HC wins	Comparable	
1	45	17	58	
Measure 2	HCPC is acceptable but HC is not	HC is acceptable but HCPC is not	Both acceptable	Both unacceptable
	25	4	91	0

## 5. Conclusions and future work

In this paper, we propose a projective clustering algorithm HCPC (*Hill-Climbing based Projective Clustering*) based on the framework of EPCH and the idea of hill-climbing algorithm for color image segmentation. It not only avoids the problem of using histogram to estimate the local data distribution in EPCH, but also extends the hill-climbing algorithm with the ability of finding clusters in subspaces. Moreover, our algorithm processes images directly at pixel/object level, resulting in an overall better segmentation than histogram-based algorithms, and is more scalable than hill-climbing algorithm when the dimensionality of the feature space is high.

The current algorithm still has a lot of room for improvement. For example, we simply set a similarity threshold to merge similar signatures, and use a minimum cluster size requirement to remove trivial clusters. In future work, we also want to consider the spatial relationship of two clusters when calculating their similarity and deciding whether or not to merge them. HCPC is essentially a general projective clustering algorithm. The criterion that is used to remove trivial clusters may not be the optimal strategy in image segmentation, because one cluster may contain several spatially disconnected segments. One possible improvement is to remove trivial segments rather than removing trivial clusters.

## 6. Acknowledgement

This work was supported in part by NSF IOS-1127911.

## 7. References

- [1] L. G. Shapiro and G. C. Stockman, *Computer Vision*, Prentice-Hall, New Jersey, pp. 279-325, 2011.
- [2] J.D. Osborne, S. Gao, W.-B. Chen, A. Andea, and C. Zhang, “Machine Classification of Melanoma and Nevi from Skin Lesions,” *Proc. of ACM Symposium on Applied Computing (SAC)*, pp. 100-105, Mar. 21-24, Taichung, Taiwan, 2011.
- [3] T. Ohashi, Z. Aghbari, and A. Makinouchi, “Hill-climbing Algorithm for Efficient Color-based Image Segmentation,” *Proc. of IASTED Intl. Conf. On Signal Processing, Pattern Recognition, and Applications (SPPRA)*, Jun. 30-Jul.2, Rhodes, Greece, 2003.
- [4] E.K.K. Ng, A. Fu, and R.C.W. Wong, “Projective Clustering by Histograms,” *IEEE Trans. on Knowledge and Data Engineering (TKDE)*, Vol. 17, No. 3, pp. 369-383, March 2005.
- [5] S. Gao, C. Zhang, and W.-B. Chen, “Identifying Image Spam Authorship with a Variable Bin-width Histogram-based Projective Clustering,” *Proc. of IEEE International Conf. on Multimedia and Expo (ICME)*, pp. 1-6, Jul. 11-15, Barcelona, Spain, 2011.

[6] C. C. Aggarwal and P. S. Yu, "Finding generalized projected clusters in high dimensional spaces," Proc. of the 2000 ACM SIGMOD Intl. Conf. on Management of Data, pp. 70-81, May 16-18, Dallas, TX, USA, 2000.

[7] Scott D. W., *Multivariate Density Estimation: Theory, Practice, and Visualization*, Wiley & Sons, 1992.

[8] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Scene Parsing with Multiscale Feature Learning, Purity

Trees, and Optimal Covers," CoRR abs/1202.2160: (2012).

[9] D. Duvenaud, "Multiscale Conditional Random Fields for Machine Vision," M.S. thesis, Dept. of Computer Sciences, The University of British Columbia, Vancouver, Alberta, 2010.



**Figure 6.** Some sample segmentation results of the proposed algorithm (HCPC), REVBH and Hill-Climbing algorithm (HC) on  $HSV_rV_gV_b$  space