OBJECT RECOGNITION FROM RESIDUAL VECTOR QUANTIZATION GENERATED FINE-GRAINED SEGMENTATION MAPS

Syed Irteza Ali Khan, Student IEEE Member
Christopher F. Barnes, Senior IEEE Member
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332, USA
irtezaa@gatech.edu
chris.barnes@gtsav.gatech.edu

ABSTRACT

The aim of this paper is to analyze the theory of direct-sum residual vector quantization (RVQ) for its application in pattern recognition. Residual vector quantization, with its direct-sum multistage codebook structure, provides a suitable framework for partitioning a high-dimensional space with a very dense covering. The direct-sum nature of the codebooks incurs relatively low (linear rather than exponential) computational and memory costs and, thus, makes an interesting candidate technique for performing very fine-grained segmentation of high dimensional input space based on pixel block queries. Markov random field (MRF) matched to the direct sum structure provides an elegant Bayesian framework to define class-conditional probabilistic models over class-specific segmented regions, generated by RVQ, and thus can effect a multi-class classification solution optimal in the *maximum aposteriori probability* (MAP) sense. Moreover, the MRF model can incorporate inter-stage dependencies in the multi-stage structure of RVQ, which is expected to improve the classification-performance. Support vector machine (SVM) is another effective technique for designing class-regions consisting of the class-specific *Voronoi* regions, generated by RVQ.

KEYWORDS: Residual vector quantization (RVQ), Markov random field (MRF), fine-grained classification, support vector machines SVM), fine-grained segmentation

INTRODUCTION

Residual vector quantization (RVQ) [1] has been used for image-driven data mining to detect features and objects in a digital image with a degree of success [2],[3]. Figure 1 illustrates the construction of a direct-sum RVQ [1] (also referred to as RVQ in this paper) with *N* stages and two code-vectors per stage wise codebook. The stages are numbered in the top-down manner, where the first stage is the top-most layer and the last stage is the bottom-most layer of the RVQ. RVQ methods based on σ-tree structures have been designed to implement successive refinement of information for image segmentation. In such implementations, novel RVQ based methods are devised for pixel-block mining, pattern similarity scoring, class label assignments and attribute mining. Direct sum σ-tree structures are used for near-neighbor similarity scoring. The variable bit-plane data representations produced by residual vector quantization based on σ-tree structures not only provide an approach for image content segmentation and a structure for formulation of Bayesian classification, but also offer a solution to the challenge of high computational costs involved in pixel-block similarity searching. Such σ-tree based multi-stage RVQ classifiers have yielded promising image-content segmentation/classification based on fine-grained features produced by the lower level feature extraction classifier.

A Gibbs random field (GRF) provides a joint probabilistic framework to model the object identification task in digital images. However, solving the joint probabilistic image model proves to be intractable. The Hammersely-Clifford theorem establishes the Markov-Gibbs equivalence whereby the GRF can be equivalently represented by a Markov random field (MRF) [9], [10]. This powerful theorem allows the global property of the GRF to be broken down to an MRF with local property. As a result, the image segmentation task can be solved optimally in the maximum posteriori probabilistic sense. The MRF can be effectively used to optimally perform the task of the object identification combined with the σ-tree based RVQ in the maximum posteriori probabilistic sense. Thus, the advantages of the σ-tree based RVQ classifier to provide fine-grained features of the object of interest can be exploited with an MRF-based model of the object.

It is observed that in images containing urban scenes, data of various classes like 'Vehicle', 'Building', 'Road', 'Pavement', 'Tree' and 'Grass' etc., are highly intermixed. An example of such data is shown in Figure 2. The input image, Figure 4a, is a traffic scene of Tokyo, Japan, obtained from Google Earth images. For the purpose of data visualization, principle component analysis is carried out on the RVQ coded training data, extracted from the input image. The first three principle components (PC) represent approximately 55% information of the original *N-tupple* data, for $N = 8$. Therefore, the first three PCs give a fair idea of the distribution of the RVQ-coded training data. The data projected into the subspace of the first three principle components, Figure 2, gives a fair idea of the nature of the data where different class- type data are much

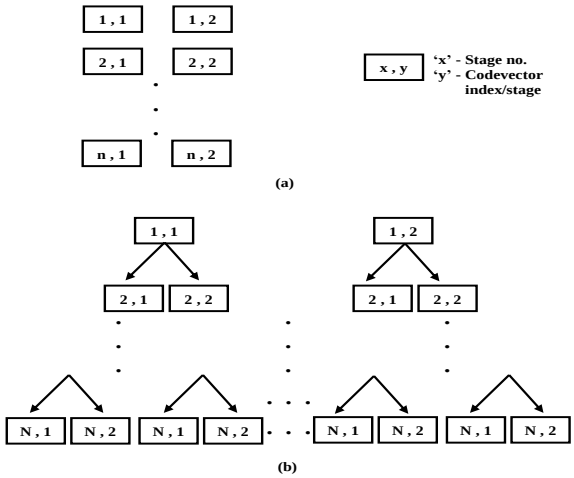


Figure 1. (a) 2-code-vector/stage RVQ with n=N stages or codebooks. (b) Direct sum codebook for N-stage RVQ with 2-code-vectors/stage.

intermixed. The ability of RVQ to generate localized, dense tessellations over disconnected sub-regions of the input space is suitable for classification in such images. RVQ has been exploited in object detection for applications such as urban area scene understanding, disaster management, etc. [2],[3]. However, in the context of object identification the methods have been applied heuristically on single stages of the multi-stage σ-tree based RVQ. As a result, higher order object detection with optimal rejection of false alarms has not yet been achieved.

In addition to MRF, support vector machines (SVMs) [14] are also very effective in defining class-specific regions over the tessellations generated by RVQ on the high-dimensional input space. These two methods will be discussed in the following section. The use of MRF framework on the direct-sum RVQ is termed as *MRF-RVQ* and *SVM-RVQ* denotes the application of SVM on direct-sum RVQ structure for the purpose of classification.

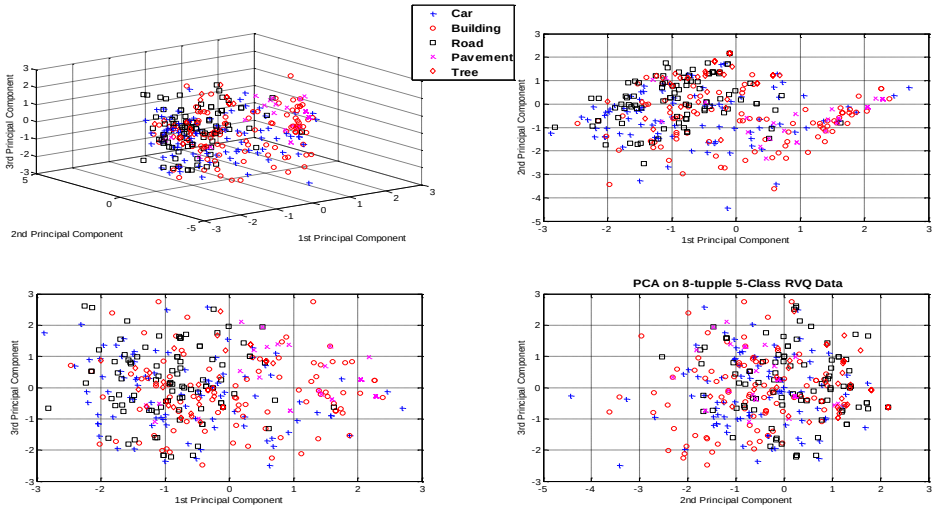


Figure 2. Principle Component Analysis of RVQ coded training data from four classes i.e., 'Vehicle', 'Building', 'Road & Pavement', 'Tree & Grass'. **Top-Left:** 3-D plot for first three principle components. **Top-Right:** 2-D plot between the 1st and 2nd principle components. **Bottom-Left:** 2-D plot between the 1st and 3rd principle components. **Bottom-Right:** 2-D plot between the 2nd and 3rd principle components.

CLASSIFICATION SCHEMES FOR RESIDUAL VECTOR QUANTIZAION

Bayesian Labeling Based on Markov Random Field (MRF)

Bayesian labeling based on MRF has been for object matching under contextual constraints [9]. Let $l = \{1,2,\dots,m\}$ be a set of discrete sites and $L^+ = \{1,2,\dots,M\}$ be a set of labels which include *M* physical labels (1,2,….,*M*) and a virtual *NULL* label (0). The aim is to assign a label from L^+ to each of the sites in l, subject to some contextual constraints. Let $f = \{f_1, f_2, \dots, f_m\}$ be a configuration of an MRF with $f_i \in L^+$ assuming a mapping $f : l \rightarrow L^+$ or a labeling of l. Let $W = L^+ \times L^+ \times \dots \times L^+$ (*m*-time) be the set of all possible configurations.

Given the likelihood function $p(r \mid f)$ and a priori probability $p(f)$, the posterior probability can be computed by using the Bayesian rule $P(f|r) \propto p(r|f)P(f)$. The Bayesian labeling problem is the following: given the observation *r*, find the map configuration *f** from an admissible space *w*, that is,

$$f^* = argmax_{f \in W} P(f|r)$$

According to the *Hammersley-Clifford* theorem of *Markov-Gibbs* equivalence [20],[21],[23], the prior probability $p(f)$ obeys a gibbs distribution

$$P(f) = \mathrm{Z}^{-1} \times e^{-\frac{1}{\mathrm{T}}\mathrm{U}(f)}$$

Where *Z* is a normalizing constant, *T* is a global control parameter called the temperature and $U(f)$ is the prior energy. The prior energy has the form

$$U(f) = \sum_{c \in C} V_c(f)$$

Where *c* is the set of cliques in a neighborhood system $\boldsymbol{\mathcal{N}} = \{\boldsymbol{\mathcal{N}_i} | \forall_{\boldsymbol{i}} \in \boldsymbol{l}\}$ for *l* in $\boldsymbol{\mathcal{N}_i}$ is the collection of sites neighboring to *i*. The likelihood $p(r|f)$ depends how *r* is observed. It can usually be represented in an exponential form $\mathrm{p}(\boldsymbol{r}|\boldsymbol{f}) = \boldsymbol{Z^{-1}} \times \boldsymbol{e^{-U(r|f)}}$, where $\boldsymbol{U}(\boldsymbol{r}|\boldsymbol{f})$ is the likelihood energy. Hence the posterior probability is Gibbs distribution $p(\boldsymbol{f}|\boldsymbol{r}) = \mathbf{Z^{-1}} \times \boldsymbol{e^{-\mathrm{U}(f|r)}}$ with posterior energy

$$U(f|r) = U(f)/T + U(r|f)$$

The likelihood energy $U(r|f)$ is given by

$$U(r \mid f) = \sum_{i \in l; f_i \neq 0} V_1(r \mid f) + \sum_{i,j \in l; f_i, f_j \neq 0} V_2\left(r \mid f_i, f_j\right)$$

Therefore, given an observation *r*, a labeling *f* of sites in *l* is also an MRF on *l* with respect to $\boldsymbol{\mathcal{N}}$. The prior energy $\boldsymbol{U}(\boldsymbol{f})$ can be ignored since all the sites are equi-probable. Therefore, the map solution is equivalently found by

$$f^* = argmin_{f \in W} U(r|f)$$

MRF-RVQ Classifier

In the *MRF-RVQ* structure, the MRF is implemented as a relational graph (RG) on the stages of the RVQ. The stages are indexed by a set $\mathcal{P} = \{1,2,\dots,P\}$ and the *codevectors* at a given stage of the RVQ are indexed by a set $\mathcal{M} = \{1,2,\dots,M\}$. The RG has the stage-wise codebooks or the constituent stage-wise code-vectors as nodes. In MRF-RVQ classification, the MRF model is defined only over the shortest-search path taken by the RVQ. The search criterion in the RVQ is stage-wise nearest-neighbor rule. At every stage the MRF-RVQ classifier has only one node $i \in \mathcal{M}$ instead of *M* nodes.

MRF-RVQ model is characterized by the associated clique structure, clique parameters and the neighborhood system need to be defined. The clique parameters, to be estimated, are ${}^{*}P_{**}$ which are the weights associated with the binary relations between the code-vectors in a pair-wise clique. These clique parameters determine which binary relations between the code-vectors in a clique get more importance in a particular class-conditional MRF-RVQ model. The clique parameters are estimated by calculating the frequency of the mapping of training inputs to the individual code-vectors in the stage-wise codebooks. Intuitively, the stage-wise mapping within the code-vectors of the MRF-RVQ model can be considered as a probability transition cycle between the stage-wise codebooks and the pair-wise clique parameters are, then, the transitional

probabilities between the code-vectors in the pair-wise cliques. Every possible binary relation in a pair-wise clique is weighted with the estimated clique parameter ${}^{k}P_{ij}$ where k is the index of the stage, i={1,2} and j={1,2} are the indices of the code-vectors of the stage-wise codebook such that the ${}^{k}P_{ij}$ indicates the transitional probability from the i^{th} code-vector of the *k-1* stage-wise codebook to the j^{th} code-vector of the codebook at k^{th} stage of the MRF-RVQ. Moreover, ${}^{1}P_{01}$ is defined as the probability of the input to map to the first code-vector of the codebook at the first stage of MRF-RVQ. Similarly, ${}^{1}P_{02}$ is the probability of the input to map to the second code-vector of the codebook at the first stage of MRF-RVQ.

The binary relations $V_2\big(r \mid f_i, f_j\big)$ and the corresponding class-conditional posterior energy $U_{l\in\mathcal{L}}(f|r)$ are defined are as under:-

$$V_2\big(r_2(i,j)\big|f_i,f_j\big) = [r_2(i,j) - R_2\big(f_i,f_j\big)]^2/2\sigma_2^2$$

Since only pair-wise cliques are considered and the label set $\mathcal{L}$ does not contain the *NULL* class i.e., $f_i \neq 0$, $U(f|r)$ can be re-written as: -

$$U_{l\in\mathcal{L}}(f|r) = -[\sum_{\{i,j\}\in C; f_i = f_j = l\in\mathcal{L}} \alpha V_2\,(r_2(i,j) \mid f_i, f_j)\,]$$

$$\alpha_l \equiv \text{Clique Parameter } {}^{k}P_{ij} \text{ corresponding to class } 'l'$$

$r_2\,(i,j)$ is a random variable. It is denoted as the MSE distance between pair-wise code-vectors i and j. $R_2\big(f_i,f_j\big)$ is the mean of the random variable $r_2\,(i,j)$ and σ_2^2 is the variance. Each input at a node $i \in \mathcal{M}$ is assigned a label from the set $\mathcal{L} = \{1,2,\dots,L\}$ according to the *maximum aposteriori probability (*MAP) rule. The set of pair-wise cliques C contains the cliques along only the shortest-search path taken by the RVQ according to the stage-wise nearest-neighbor rule.

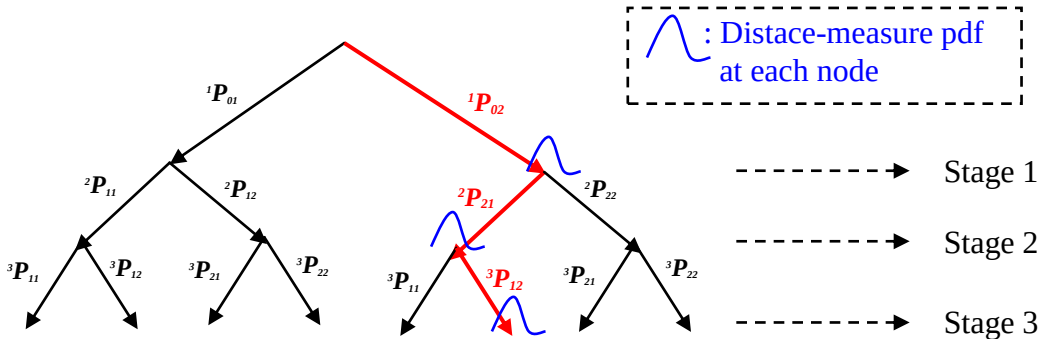


Figure 3. MRF-RVQ multi-classification scheme using *MRF-RVQ* classifier.

A class-conditional MRF is characterized by its clique parameters ${}^{k}P_{ij}$. In other words, each class has its own set of corresponding transitional probabilities. As an example, in Figure 3, the red path consisting of three inter-stage branches, from stage-1 to stage-3, is the path taken by a test input. The associated clique parameters for the pair-wise binary relations are the probabilities ${}^{k}P_{ij}$, which are also colored red. The class decision $l^* \in \mathcal{L}$ is

$$\begin{aligned} l^* &= \ argmin_l(U_l(r/f) \\ &= \ argmax_l\Big(\ {}^{1}P_{02}V_2(r_2(0,2) \mid f_0,f_2) + \ {}^{2}P_{21}V_2(r_2(2,1) \mid f_2,f_1) + \ {}^{3}P_{12}V_2(r_2(1,2) \mid f_1,f_2)\Big) \end{aligned}$$

The performance of the proposed MRF-RVQ classifier is shown in Figure 4. The *m*-ary classification result, for *m*=5, is shown in Figure 4(d) for the input image shown in Figure 4(a). The output classification map in Figure 4(b) is generated by using all eight stages of the RVQ. The specifications of the direct-sum codebook RVQ are listed as under:-

a) Number of Stages $P = 8$.
b) Number of *codevectors* per stage $M = 4$.
c) Number of classes $m = 5$.
d) Size of direct sum code book = $M^P = 4^8$

= 65,536 *codevectors.*

e) Size of *codevector*/template = 13x13 pixels.
f) Number of training vectors per class = 100 vectors

The five classes represent the following objects: -

Class 1 = 'Vehicle' (dark-blue color)
Class 2 = 'Building' (light-blue color)
Class 3 = 'Road' (brown color)
Class 4 = 'Pavement' (orange color)
Class 5 = 'Trees and Grass' (green color)

For *m*=4 class-types 'Road' and 'Pavement' are combined into one class-type, which is color-coded orange, Figure 4(b).

As shown in the Figure 4(b,d), the candidate areas for the particular classes are identified. For example, the vehicles are identified (light-blue color for $m = 4$ and dark-blue color for $m = 5$). It is encouraging to note that despite the color and size variation among the vehicles, their detection is very effective. Similarly, edges of the detected vehicles have also been successfully detected and color coded as the type 'Vehicle'. Similarly, edges of other structures such as buildings and roads are also detected. Moreover, the road sections and the pavements sections, which are combined into one class for $m = 4$ [orange color-coded in Figure 4(b)], are separately classified for $m = 5$ ['Road' is color-coded brown and 'Pavement' is orange colored in Figure 4(d)], thus, producing a finer classification map. This result is encouraging towards attaining fine-grained classification.

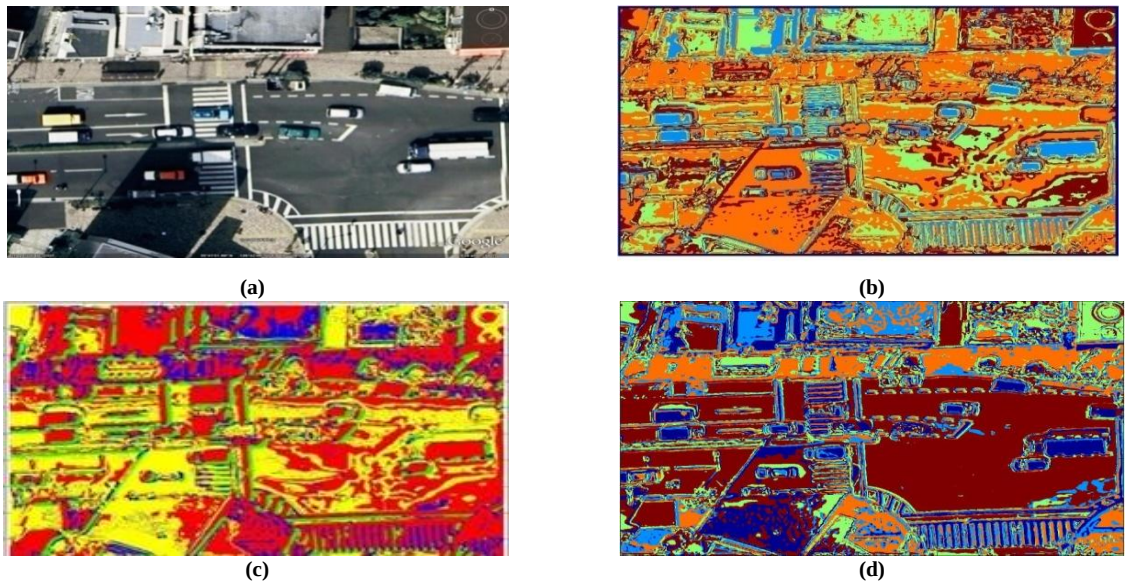


Figure 4. Multi-classification results for *MRF-RVQ model 2*. **(a)** The input image *(Courtesy Google Earth)*. **(b)** Color-coded classification map for 4 classes: 'Vehicle' (light blue), 'Building' (green), 'Road & Pavement' (orange), 'Tree & Grass' (brown). **(c)** RVQ-generated segmentation map at the 1st stage of RVQ. **(d)** Color-coded classification results for 5 classes: 'Vehicle' (dark blue), 'Building' (light blue), 'Road' (brown) & Pavement' (orange), 'Tree & Grass' (green).

SVM-RVQ Classifier

The purpose of employing MRF on RVQ is to generate class-regions over the input space. Support vector machines (SVM) [14] are another technique for defining optimal class-regions over the input space. SVMs have been successfully used for pattern recognition. Intuitively, given a set of points which belong to either of two classes, a linear SVM finds the *hyperplane* leaving the largest possible fraction of points of the same class on the same side, while maximizing the distance of either class from the hyperplane. The hyperplane is determined by a subset of the points of the two classes, named support vectors. Nonlinear SVM is most relevant in our research as, in our case, the

data from different classes is not only non-separable but also much intermixed. In nonlinear SVM, the original data is transformed to an enlarged space. Generally, the linear boundaries in the enlarged space can perform better separation of class-specific training data. [11],[12],[13] provide comprehensive details on nonlinear SVM.

The classification map generated by a 5-class SVM-RVQ is shown in Figure 5(c). 3rd-order polynomial kernel is used in the training of the SVM. The training data for the SVM is the *P-tupple* representation of the vectors that are used to train the RVQ. The RVQ is designed with $P = 8$ stages, with $M = 4$ *codevectors* per stage. The SVM is applied on the direct-sum *codevector* space of the RVQ at the end of the 8th stage. The specifications of the direct-sum codebook RVQ are listed as under:-

a) Number of Stages $P = 8$.
b) Number of *codevectors* per stage $M = 4$.
c) Number of classes $m = 5$.
d) Size of direct sum code book = $M^P = 4^8$
   = 65,536 *codevectors.*
e) Size of *codevector*/template = 13x13 pixels.
f) Number of training vectors per class = 100 vectors

The five classes represent the following objects: -

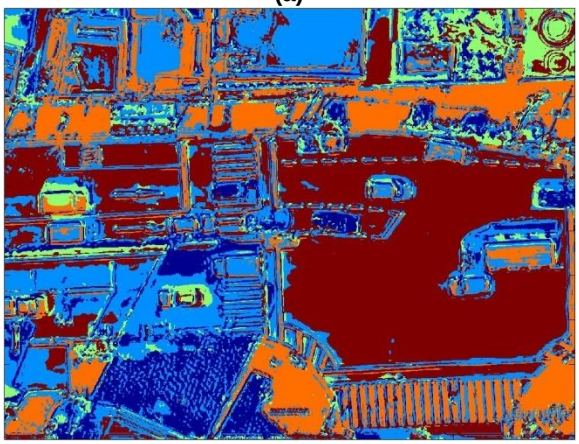
Class 1 = 'Vehicle' (dark-blue color)
Class 2 = 'Building' (light-blue color)
Class 3 = 'Road' (brown color)
Class 4 = 'Pavement' (orange color)
Class 5 = 'Trees and Grass' (green color)

When compared with SVM-only classifier, Figure 5(b), SVM-RVQ classifier has performed better. It can be seen that in the lower-left area of Figure 5(a), the distinction of the road and pavement sections is much better in the SVM-RVQ classification map, comparatively. The SVM-only classifier has falsely classified significant portion of the road section in the lower-left area as the class-type 'Building', whereas, the pavement area is largely falsely labeled as the class-type 'Vehicle'. It should be noted that the two classifiers are trained on different types of data. The SVM-only classifier is trained on the training vectors of the RVQ, whereas, the SVM-RVQ classifier is trained on the *P-tupple* representation of the same training vectors. The improved performance of the SVM-RVQ classifier encourages us to use RVQ to transform the original data into the corresponding *P*-tupples and, then, use the transformed data or, equivalently, the *P-tupple* representation of the original data to perform classification on.

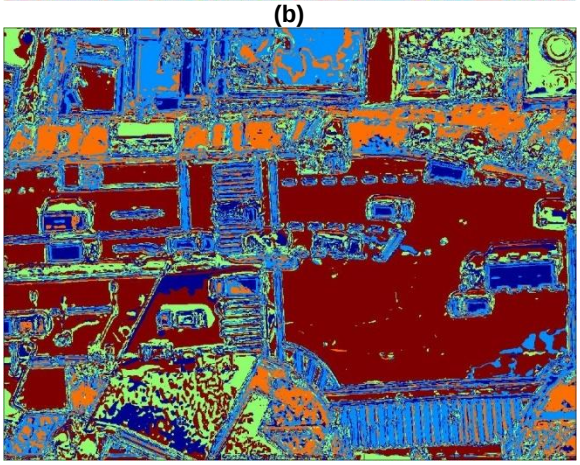
CONCLUSION

The ability of RVQ to generate localized, dense tessellations over disconnected sub-regions of the input space is suitable for fine-grained segmentation of images. It has been shown that MRF and SVM are suitable techniques to define defining optimal class-regions over the same class-type connected or even unconnected regions, covering the input space. SVMs are effective tools to generate optimal class-regions in the direct-sum space of the input, whereas, MRF provide a robust framework to incorporate the inter-stage decencies of data generated by the direct-sum RVQ. The interplay of MRF and SVM schemes on the multi-stage structure of RVQ seems potentially a approach to fine-grained classification based on the fine-grained segmentation maps of the RVQ.

(a)



(b)



(c)

Figure 5. *SVM-RVQ* classification results for 5 classes. **(a)** Input image of Tokyo traffic scene (*courtesy Google Earth*). **(b)** SVM-only classification map. **(c)** *SVM-RVQ* classification map.

REFERENCES

[1] C. F. Barnes, "Vector quantizers with direct sum codebooks," *IEEE Trans. Info. Theory*, vol. 39, issue 2, Mar. 1993 pp. 565 – 580.

[2] C. F. Barnes, "Image-driven data mining for image content Segmentation, classification, and attribution," *IEEE Trans. Geoscience and Remote Sensing*, vol. 45, issue 1, Sept. 2007 pp. 2964 – 2978.

[3] C. F. Barnes, H. Fritz, Jeseon Yoo, "Hurricane Disaster Assessments With Image-Driven Data Mining in High-Resolution Satellite Imagery," *IEEE Trans. Geoscience and Remote Sensing*, vol. 45, issue 6, part 1, June 2007 pp. 1631 - 1640.

[4] B. H. Juang, and A. H. Gray,"Multiple stage vector quantization for speech coding," *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing*, vol. 7, May 1982, pp. 597-600.

[5] J. Makhoul, S. Roucos, and H. Gish,"Vector quantization in speech coding," *Proc. IEEE*, vol. 73, issue 11, Nov. 1985, pp.1551-1588.

[6] D. V. Arnold,"Vector quantization of synthetic array radar data," Master's thesis, Brigham Young Univ., Provo, UT, 1987.

[7] W.Y. Chan, S Gupta, and A. Gresho, "Enhanced multistage vector quantization by joint codebook design," *IEEE Trans. on Comm*, vol. 40, No. 11, Nov. 1992 pp. 1693-1697.

[8] C. F. Barnes, S. A. Rizvi, and N. M. Nasrabadi,"Advances in residual vector quantization: A review," *IEEE Trans. Image Proc*, vol. 5, issue 2, Feb. 1996 pp. 226-262.

[9] S. Z. Li,"A Markov random field model for object matching under contextual constraints," *Proc. IEEE CVPR'94*, Jun. 1995, pp. 866-869

[10] G. Geman and D. Geman, "Stochastic relaxation, gibbs distribution and bayesian restoration of images", *IEEE Trans. Paterrn Analysis and Mach. Intelligence*, PAMI (6): 721-741, Nov. 1984

[11] Burges, Christopher J.C. "A Tutorial on *Support* Vector Machines for Pattern Recognition", *Data Mining and Knowledge Discovery, 2, 121–167, 1998.*

[12] Bishop, C.M. "Pattern Recognition and Machine Learning", *© 2006 Springer Science + Business Media, LLC.*

[13] Theodoridis, S., and Koutroumbas, K. " Pattern Recognition", *3rd edition, © 2006, Elsevier (USA).*

[14] Corinna Cortes and V. Vapnik, "Support-Vector Networks", *Machine Learning, Vol. 20, pp 273-297, 1995.*