Characterizing Colonic Detections in CT Colonography Using Curvature-Based Feature Descriptor and Bag-of-Words Model

Javed M. Aman, Ronald M. Summers, and Jianhua Yao

Radiology and Imaging Sciences Department, Clinical Center,
National Institutes of Health, Bethesda, MD, USA

Abstract. We present a method based on the content-based image retrieval (CBIR) paradigm to enhance the performance of computer aided detection (CAD) in computed tomographic colonography (CTC). The method explores curvature-based feature descriptors in conjunction with bag-of-words (BoW) models to characterize colonic detections. The diffusion distance is adopted to improve feature matching and clustering. Word selection is also applied to remove non-informative words. A representative database is constructed to categorize different types of detections. Query detections are compared with the database for classification. We evaluated the performance of the system by using digital phantoms of common structures in the colon as well as real CAD detections. The results demonstrated the potential of our technique for distinguishing common structures within the colon as well as for classifying true and false-positive CAD detections.

Keywords: CAD, CT colonography, affine invariant feature, bag-of-words.

1 Introduction

Cancer screening and early detection are an important step in colon cancer prevention. Optical colonoscopy (OC) is the traditional colon cancer screening procedure. However, because of its invasiveness, many patients forego this procedure. Computed tomographic colonography (CTC) has emerged as a minimally invasive screening procedure. CTC can benefit from CAD systems to improve the sensitivity and reduce the interpretation time [1]. Most CAD systems require post processing to reduce the number of false positives. We propose a method based on the Content-Based Image Retrieval (CBIR) paradigm to enhance the CAD performance.

CBIR is a computer vision technique for searching for similar images within an image database. It has been used in applications such as medical image searching [2] and artwork retrieval [3]. The images in a CBIR system are characterized as a set of feature descriptors computed directly from the images. Detecting affine transformation-invariant salient feature points in an image is important for the success of a CBIR system. The scale-invariant feature transform (SIFT) proposed by Lowe [6] is one of such feature descriptors. However, images vary greatly in the number of

H. Yoshida et al. (Eds.): Virtual Colonoscopy and Abdominal Imaging 2010, LNCS 6668, pp. 15–23, 2011.

feature points, which makes the comparison of two images difficult. The subsequent classification also requires a feature vector of fixed dimension. A vector quantization (VQ) technique was proposed to handle this problem. The bag-of-words (BoW) model [4] was one of the VQ techniques and was first introduced in natural language processing and then in computer vision and information retrieval, especially for object categorization. In the BoW model, feature points are grouped into clusters or "words" that represent the specific feature pattern shared by all feature points in the clusters. By mapping of its feature points into words, an image can be represented as a "bag of words" which can be employed in further classification.

In this paper, we propose a SIFT-like feature descriptor based on curvatures and incorporate it with the BoW model in a CBIR framework. Our method was validated with both phantom and clinical CTC data and demonstrated promising results.

2 Methods

Our system is a post-processing step for a CAD system on CTC. The CAD system segments the colon and generates a set of potential polyp detections based on local curvature and CT attenuation. Post-processing steps (such as support vector machines and CBIR) then further filter the detections to reduce the number of false positives. In our CBIR framework, the detections are cropped from the original CTC images by use of their segmentation boundaries. The cropped images are then re-sampled to uniform 64*64*64 blocks by use of B-Spline interpolation. Affine-invariant feature points are extracted from the image, and BoW models are generated. A database is constructed that stores representative detection images and their associated BoW models. A new detection is then compared against the database, and the retrieval results are employed to make a classification decision.

2.1 Curvature-Based Feature Descriptor

Feature points are salient points in the image that contain rich local image information. It is desirable that the feature descriptor is affine-invariant, so that similar images in different poses and scales present similar features. Features used in our method are derived from the n-dimensional scale-invariant feature transform (N-SIFT) method proposed by Cheung et al. [5], which was generalized from the 2D SIFT originally proposed by Lowe [6]. The method is comprised of two steps: feature point detection and feature descriptor generation.

In SIFT, feature points are related to the extrema points in the image's gradient space. A set of Gaussian smoothing filters (at different sigma scales) is applied to the image to generate a set of difference of Gaussian (DoG) images. Pixels that are extrema in their surrounding 3x3x3 neighborhoods in DoG are preliminary feature points. Duplicate points found in multiple DoG images are trimmed, leaving only the points with the greatest magnitude. The feature points are detected in a multi-scale image pyramid. The image in the successive scale is a linearly interpolated,

downsampled version of the Gaussian smoothed image in the previous scale. Feature points are detected in each scale independently, and their positions are then restored to the first scale. Figure 1b and 1d show feature points detected on a true-positive and false-positive image from CTC.

In the original SIFT implementation, the feature descriptor is constructed from a local image gradient that is weighted by the distance to the feature point. Cheung et al. [5] showed that gradient-based features can only cope with up to 10° rotation variation. In order to increase the robustness to the rotation, we propose a curvature-based feature descriptor. The shape index describes the local surface shape and is computed from the principal curvatures captured by a local Hessian matrix,

$$s = \frac{2}{\pi} \arctan \frac{\kappa_2 + \kappa_1}{\kappa_2 - \kappa_1} \tag{1}$$

Here κ_1 and κ_2 are the principal curvatures, and s is the shape index. The value ranges from 0.0 to 1.0, which corresponds to concavities and convexities such as ruts, troughs, caps, domes, and ridges [7]. The shape index is both scale- and rotation-invariant. A histogram of the shape index in a 9x9x9 neighborhood of the feature point is used as the feature descriptor. The histogram is binned from 0.0 to 1.0 at 0.05 intervals, for 20 bins.

The similarity between two feature descriptors can be evaluated by their distance. The most straightforward is the Euclidean distance, where a bin-to-bin distance is summed up. However, the Euclidean distance does not take into account the relationship between neighboring bins and may suffer from a rounding effect when assigning the histogram. To handle this situation, we apply a diffusion distance metric for comparison of our feature descriptors [8]. The diffusion distance is a cross-bin comparator of histograms. It models the difference between two histograms as a temperature field and considers the diffusion process on the field. A Gaussian pyramid scheme is implemented to discretize the continuous diffusion process and the sum of the norm over all of the pyramid layers. The diffusion distance is computed as follows. First, the difference between the two histograms is set as the first layer of the pyramid. The next layer is the downsampled (by two) histogram of the previous layer convolved with a Gaussian filter (σ=0.5). This process is repeated until there is only one bin in the histogram. The sum of the differences in all layers is the diffusion distance.

$$\begin{aligned} K(h_1, h_2) &= \sum_{i=0}^{n} |d_i| \\ d_0 &= h_1 - h_2, \quad d_i(x) = d_{i-1}(x)_{\downarrow 2} * \varphi(x, 0.5) \end{aligned} \tag{2}$$

Here K is the diffusion distance, h_1 and h_2 are two feature descriptors, d_0 is the difference of the histograms at the first layer, d_i is the downsampled version of d_{i-1} and φ is a Gaussian function.

2.2 Bag-of-Words (BoW) Model

After the feature points are extracted, the detection image can be characterized by a set of feature descriptors. We then employ the vector quantization (VQ) technique [4] to generate a codebook from the feature descriptors. The descriptors are clustered by use of the K-means clustering algorithm [9], and the center of each cluster is a codeword. The codeword is indexed, and the histogram of the codeword appearances in an image is used as the Bag-of-Words model and applied in the subsequent image classification.

K-means clustering has two primary pitfalls: sensitivity to initial cluster centers and high computational complexity. To overcome these, a kd-tree data structure is introduced. The kd-tree organizes the feature space orthogonally and hierarchically. It is a binary tree dividing a high-dimensional space and is constructed as follows: starting from the root, for every non-leaf node, a splitting hyperplane at the median point of the longest axis of the node divides the space into two subspaces (nodes). The splitting process is iterated until there is only one point in each node (leaf). Initial cluster centers are taken from points in nodes at the same level of the tree to ensure they are well separated. Although this does not technically solve the sensitivity to the initialization problem, it does allow better clustering as opposed to randomly selecting the seed points. The spatial separation of points also improves the performance of the K-means clustering by allowing it to ignore interactions between distant points. A filtering algorithm [9] of K-means clustering is applied to the kd-tree to obtain the cluster centers (i.e. codewords). During each iteration, the feature points are associated with their closest cluster centers, and the cluster centers are updated by their associated feature points. The process is repeated until the cluster center is stabilized. The kd-tree reduces the complexity from $O(n^2)$ to O(nlogn).

The BoW is a histogram recording the count of codeword occurrences in a particular image. Each feature point is associated with a codeword in the codebook. The association is computed as,

$$association(k,c,r) = \frac{\|c-k\|}{r \bullet |c-k|} \tag{3}$$

Here, k is a feature point, c is the codeword, and r is the radius of a codeword which describes the radius of the cluster represented by the codeword. A feature point is assigned the codeword with maximum association value, and one count of the codeword occurrence will be added to the BoW histogram. The histograms are normalized to account for the differences in the number of feature points among images. Figure 1c and 1f show the BoW histograms of the two detections in Figure 1a and 1d.

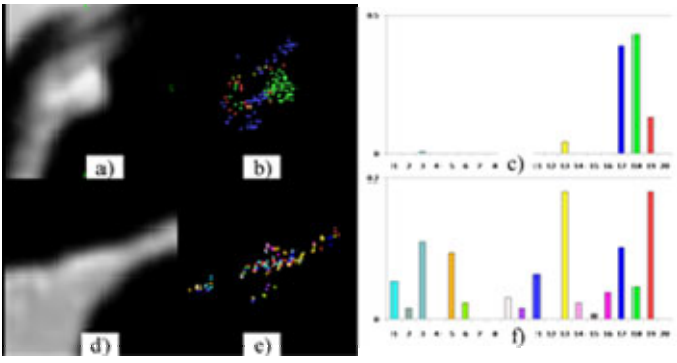

Fig. 1. Feature points and Bag-of-Words models. a) a true positive detection; b) feature points of a); c) BoW model of a); d) a false positive detection; e) features points of d); f) BoW model of d). The color code in the histogram corresponding to that for the feature points.

Not all of the codewords are useful in the object recognition. Noisy words may exaggerate the difference between similar BoW histograms. Non-informative words, such as words common across images, may skew the comparison. We apply a forward stepwise word selection scheme to choose the informative words for classification. For any word in the codebook, if removing it improves the performance, it will be dropped from the codebook, otherwise it is kept.

2.3 Content-Based Image Retrieval (CBIR)

Our system is built in a CBIR framework. We construct a database of representative detections and use it to assist classification. The database stores the detection images and their BoW histogram, and it contains equal numbers of detections in each category (TP and FP in our case). Because we have far more FP detections than TP detections in our training data, all TPs are put in the database, and FPs are randomly selected to match the number of TPs. Another strategy is to conduct a K-means clustering on the FP detections, and the centers of the clusters are used as representative detections.

Given a new detection, its BoW histogram is computed and queried against those in the database. The results are ranked by their similarities to the query, and only the top matches are retrieved. The number of the retrieval results is known as the search depth. Based on the labels (TP or FP) of the retrieval results, the attribute of the query image can be determined. Two metrics can be computed from the retrieval results. One is the TP ratio (TPr), i.e., the number of TP detections in the retrieval set divided by the search depth. The other is the normalized discounted cumulative gain (nDCG), which is a common measure of information retrieval effectiveness [10].

$$nDCG_p = \frac{\sum_{i=1}^{p} \frac{(2^{rel_i} - 1)}{\log_2(1+i)}}{\sum_{i=1}^{p} \frac{1}{\log_2(1+i)}} \tag{4}$$

Here, rel_i is the relevance variable (either 1 for a match or 0 for a non-match), p is the search depth. nDCG equals to 1 if all of the matches in the result set are relevant.

3 Experiments and Results

3.1 Phantom Experiments

Four types of realistic phantoms of common colon structures were generated: folds, walls, dents, and polyps. These were created by use of Boolean shape operators on simple shapes such as prisms, ellipsoids, and cylinders. Colon walls are modeled as the surface of cylinders with different radius. Dents are models as ellipsoids cut into the colon wall. Polyps are modeled as ellipsoids protruding from the colon wall. Folds are also modeled as thin and elongated ellipsoids. For further complication, intensity and structural noise are added to the phantoms. Intensity noise is Gaussian noise added to the pixel intensity. Structural noise is intended to add bumps and dents (in the form of 3*3*3 balls) to the colon surface.

Ten phantoms of varying size and shape of each of the four types were generated. Two noise levels, 10% intensity and 5% structural, and 20% intensity and 10% structural, were added to the phantoms. There were a total of 120 phantoms (40 clean and 40 noisy at two noise levels, respectively). Figure 2 shows examples of noisy phantom and their BoW models.

We randomly selected half of the phantoms (60) to train the codebook of 20 codewords and build the database. We then evaluated the performance by using the remaining phantoms. The retrieval depth was 15. We compared the nDCG of the system by using Euclidean distance (ed) vs. diffusion distance (dd), and no word selection (nws) vs. word selection (ws) (see Table 1).

Our method shows a strong ability to distinguish different structures in the colon. The word selection showed mixed results, marginally improving or weakening the matching of different structures. Codebooks generated by use of the diffusion distance metric show a better retrieval performance for the fold type, but a decreased performance for the wall type. The polyp phantoms showed perfect matching, because they exhibited high convex curvature feature points which were uncommon in the other structures.

3.2 CTC Experiments

We tested our method on 162 CTC studies. The CAD system based on a support vector machine resulted in 1274 detections. Of them, 94 were defined to be TPs based

on OC findings, whereas 1180 were FPs. There were 11 polyps less than 6mm, 51 between 6 and 9mm, and 32 larger than 9mm. We conducted a ten-fold cross-validation. In each run, we used nine tenths of the data to construct the representative database, and the remaining one tenth was used for testing. The number of codewords was 20. Figure 3 shows the retrieval results of the detections shown in Figure 1. Table 2 lists the mean performance of the 10-fold cross-validation. Figure 4 shows the FROC curves generated by the ROCKIT toolkit [11]. The CBIR was able to eliminate 40% of the FPs (4.3 FP per case) while maintaining the sensitivity at 91%.

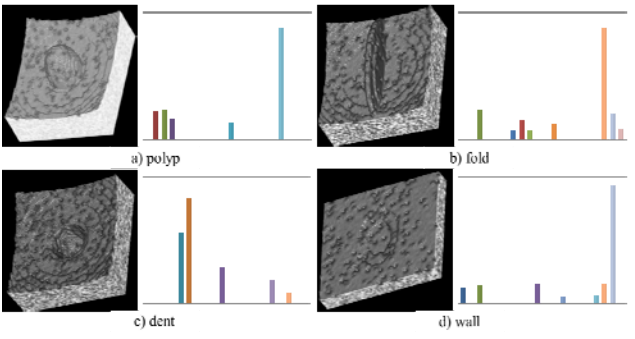

Fig. 2. Noisy phantom examples (left) and their BoW models (right)

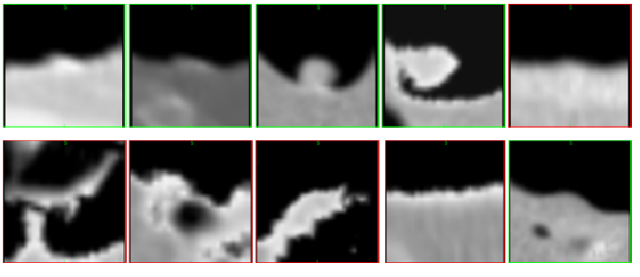

Fig. 3. Retrieval results from CBIR. First row: results of the TP detection in Figure 1a); Second row: results of the FP detection in Figure 1d). Images with green frames are TP detections, and those with red frames are FP detections.

Table 1. Summary of performance in phantom data (nDCG)

nDCG	ed-nws	ed-ws	dd-nws	dd-ws
Polyp	1.00	1.00	1.00	1.00
Fold	0.75	0.75	0.88	0.88
Dent	0.82	0.82	0.82	0.82
Wall	0.77	0.70	0.68	0.68
All	0.82	0.82	0.84	0.85

Table 2. Summary of performance in CTC data

Type	TPr	nDCG
TP	0.68 +/- 0.24	0.68 +/- 0.25
FP	0.35 +/- 0.27	0.62 +/- 0.28

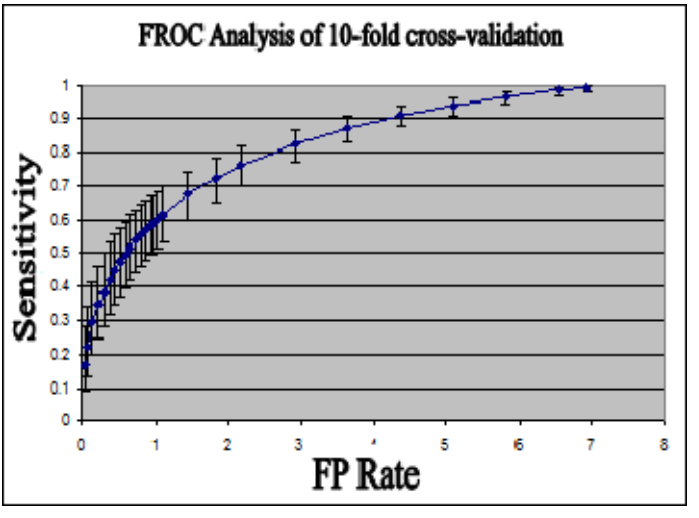

Fig. 4. FROC analysis of CTC data

4 Discussion and Conclusion

We proposed an approach to characterizing the colonic detection in CTC by using a curvature-based feature descriptor and Bag-of-Words models. The curvature-based feature descriptor provides an affine-invariant description of salient points in an image and the BoW model provides a standard platform for comparing detections. We also employed the CBIR paradigm to determine the detection attribute by using a database of pre-selected representative examples. The method was validated on both synthetic phantoms and clinical CTC data.

There is room for improvement in both the feature descriptor and the BoW model. The spatial location of the feature point can be encoded in the descriptor to assist the object recognition and image classification. Techniques developed in information retrieval such as stop word removal and various word-weighting schemes can be adopted in the BoW model. Different strategies such as document frequency, χ^2 statistics, and mutual information can also be explored in the word selection process. The CBIR technique adopted in our system is similar to the k-Nearest Neighbor (kNN) classifier. Other advanced classification techniques such as a neural network, support vector machine, and Bayes model can be applied.

References

1. Summers, R.M., et al.: Computed Tomographic Virtual Colonoscopy Computer-Aided Polyp Detection in a Screening Population. Gastroenterology 129, 1832–1844 (2005)
2. Muller, H., et al.: Review of content-based image retrieval systems in medical applications—clinical benefits and future directions. International Journal of Medical Informatics 72, 1–23 (2004)
3. Guicca, G., Schettini, R.: A relevance feedback mechanism for content-based image retrieval. Info. Proc. and Manag. 35, 605–632 (1999)
4. Yang, J., et al.: Evaluating bag-of-visual-words representations in scene classification. In: International Workshop on Multimedia Information Retrieval. ACM, Augsburg (2007)
5. Cheung, W., Hamarneh, G.: n-SIFT: n-Dimensional Scale Invariant Feature Transform. IEEE Transactions on Image Processing 18(9), 2012–2021 (2009)
6. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60, 91–110 (2004)
7. Koenderink, J.J., van Doorn, A.J.: Surface shape and curvature scales. Image and Vision Computing 10(8), 557–565 (1992)
8. Ling, H., Okada, K.: Diffusion Distance for Histogram Comparison. In: IEEE Conference on Computer Vision and Pattern Recognition, New York, USA (2006)
9. Kanungo, T., et al.: An Efficient k-Means Clustering Algorithm: Analysis and Implementation. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7), 881–892 (2003)
10. Croft, B., Metzler, D., Strohman, T.: Search Engines: Information Retrieval in Practice. Addison Wesley (2009)
11. ROCKIT, Kurt Rossman Laboratories. University of Chicago, Chicago, IL (2004)