Viewpoint Invariant Descriptor for RGB-D Images

Yan Wang
Columbia University
116th Street and Broadway
New York, NY 10027, USA
yw2383@columbia.edu

Jinyuan Feng
Columbia University
116th Street and Broadway
New York, NY 10027, USA
jf2651@columbia.edu

Deli Pei
Columbia University
116th Street and Broadway
New York, NY 10027, USA
dp2532@columbia.edu

ABSTRACT

Traditional local features already achieve resistance toward changes in illumination, translation, rotation, and scale variance, thus play an important role in visual content retrieval. However, they are not stable to 3D rotation or viewpoint changes. This report aims to solve this problem by introducing depth sensor, a new kind of sensor which can capture depth with regular optical information, in this problem and proposing a novel descriptor to achieve viewpoint invariance. We first transform input RGB-Depth images to 3D point cloud in Cartesian space, estimate normal vector and calibrate viewpoint for each local patch, and then project the cloud to 2D plane and calculate the descriptors. Some preliminary experiment results also support the effectiveness of our descriptor.

General Terms

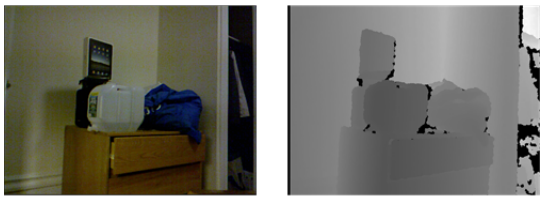
Computer vision

Keywords

3D feature, RGB-D image, Kinect, viewpoint calibration

1. INTRODUCTION

Local features like SIFT [6] and SURF [1] enable affine invariant matching among images thus play a fundamental role in visual content retrieval. However, they are still vulnerable toward 3D rotation of objects, or camera view change [8], which is actually a common problem practical retrieval system needs to solve. The emergence of new depth sensors based on laser speckle decorrelation [5] like Microsoft Kinect [7] brings an approach to accurately detect depth of each pixel in the image, i.e. the distance between each pixel and imaging plane. Fig. 1 shows a sample output which is called RGB-D image of depth sensors, containing an RGB image with corresponding depth image. With the depth info, the 3D model of objects can be inferred, which gives researchers a promising way to propose a new local feature invariant to camera view change.



(a) Optical image (b) Depth image

Figure 1: Sample output of depth sensors. On the left is optical image, and on the right is the depth image. The brightness of pixels in depth image indicates the depth of corresponding pixel in optical image.

Although depth sensors are newly emerged, similar research has been done in 3D object retrieval field, in which a fundamental problem is to compare two 3D models with camera view invariance. There are two categories of approaches. Some researchers attempt to extract local features directly from 3D models and use these features to match models. For example, Wyngaerd et. al. [13] use bitangent curves to detect interest points and use Euclidean signature and affine signature as descriptor. Some other research work projects the 3D model to lots of planes and adopts 2D techniques to realize camera view invariant, such as random projection with SIFT descriptor [9] and projecting to a cylinder to form a panorama and using DFT and wavelet responses as descriptor [10]. Wu et. al. first use Structure from Motion to get 3D model of an object from multiple photos, then project texture of the model toward the "normal directions" reasoned from each photo, and adopt SIFT descriptor to describe projected textures [12].

However, all the above approaches cannot be applied in our problem because of the following challenges. First, the output from the depth sensor is a collection of 3D points, while the direct 3D feature extraction approaches usually assume the input as mesh-based 3D models. Second, even with depth info, we cannot infer the whole 3D model from only one image. Therefore traditional projection-based approaches will get incomplete textures thus are not suitable here. Last but not least, as we will see later, the change of input from 2D image to 3D point cloud also brings a problem that the sampling density of optical camera and depth sensor is affected by camera view, which makes viewpoint

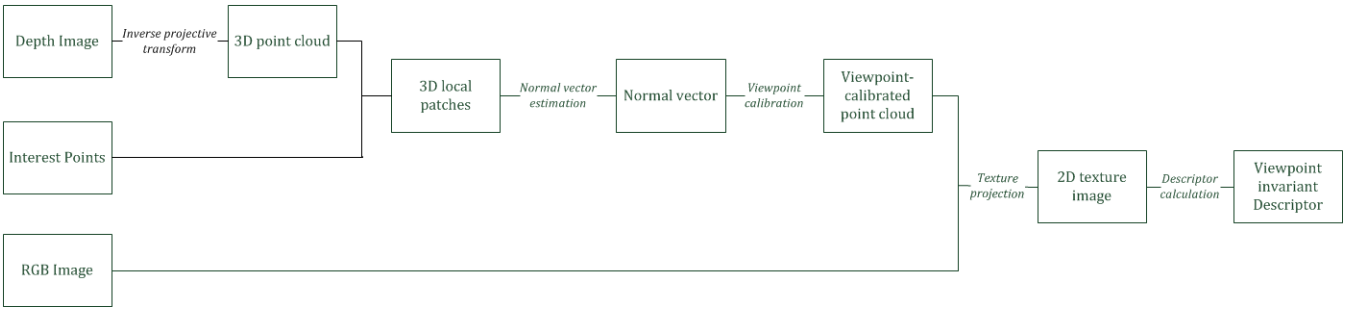


Figure 2: Framework of the algorithm.

invariant even harder to achieve.

In this paper, we provide a novel descriptor with viewpoint invariant based on RGB-D images. With an RGB-D image and detected interest points as input, we aim to get compact description of the interest points with resistance toward viewpoint changes and 2D rotations. Combined with scale invariant detectors, our descriptor can achieve 3D affine transform invariant and is expected to improve performance of visual content retrieval systems. The framework of our algorithm is, first transform the input RGB-D image to a point cloud, then estimate normal vector toward neighbor area of each interest point, do viewpoint calibration based on normal vector, project the textures into camera imaging plane, and then calculate color histogram as final descriptor.

The paper is organized as follows. Section 2 explains our algorithm. Section 3 illustrates methodology of our experiments, states results and related discussions, followed by Section 4 introducing conclusion and future directions.

2. APPROACH

2.1 Overview

Our algorithm takes an RGB-D image with detected interest points as input and outputs descriptors as 64-dimensional vectors. The algorithm framework is as Fig. 2 shows. We first do inverse projective transform according to camera's related parameters, then estimate normal vectors of each local patch. After calibrate viewpoint by rotating the local point cloud into a "standard" direction, we project the texture of the point cloud to the imaging plane and calculate descriptors.

The reason why we propose this approach based on projection is two-folds. First, projection-based methods can take advantage of textures info from RGB image and are proved effective in achieving viewpoint invariant, especially in [12]. Second, rather than projecting the whole model, our approach only projects textures of local patches, which also solves the incomplete model problem. In the following sections, we will introduce the algorithm step by step.

2.2 Point cloud reconstruction

With an RGB-D image, we need first to reconstruct the 3D model as point cloud. Considering we already have the depth/distance information, this step can also be treated as transforming the coordinate system from local camera view to global Cartesian view. That is,

$$\begin{bmatrix} x_g \\ y_g \\ z_g \end{bmatrix} = J \begin{bmatrix} x_l \\ y_l \\ z_l \end{bmatrix}$$

in which J is the Jacobian of the Cartesian coordinate system and the camera view coordinate system. x_l, y_l are the local coordinates of pixels in RGB images, and z_l is the depth of corresponding pixel. (x_g, y_q, z_q) is the coordinates of the pixel in the real Cartesian space.

2.3 Normal vector estimation

With the interest points and reconstructed point clouds, we adopt a neighbor area for each interest point. If the detector is scale invariant, i.e. every interest point p has a scale s, we adopt points in the sphere with circus s and center p as p's neighborhood. If the detector is not scale invariant, we adopt p's k-nearest neighbors as its neighborhood, in which k is a user-defined parameter.

The interest points with their neighborhoods form 3D local patches. For each local patch, we need to estimate their normal vector in order to do viewpoint calibration. Here we employ the approach from [11], which is proved to be effective and stable to noise [3]. We do a linear regression on the local patch to get a 2D plane $z = \theta^T X$ to minimize the weighted loss function

$$R(\theta) = \sum_{k=1}^{n} w(z_k)(\theta^T X - z_k)$$

A Gaussian function is used as weight function $w(z_k)$ here, which grants more weight to points close to the center but less weight to far away points. The normal vector τ of the local patch is set as the normal vector of the regressed plane.

2.4 Viewpoint calibration

With knowing the normal vector of each local patch, the core procedure to achieve viewpoint invariant is to calibrate the viewpoint of local patches to a "standard" direction. We choose the "standard" direction as $[0, 0, -1]$, i.e. exactly opposite the normal vector of the local patch. To accomplish this calibration, rotation matrix T is introduced. If setting the coordinate after calibration as $X = [x, y, z]^T$ and original coordinate from Section 2.2 as $X_0 = [x_0, y_0, z_0]^T$, the transform process can be expressed as

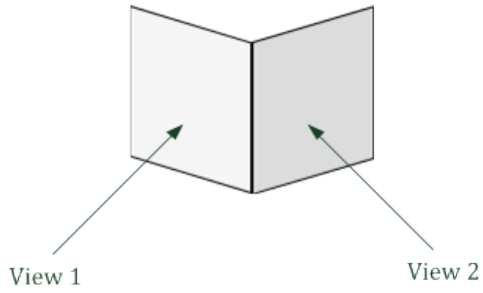
$$X = TX_0$$

in which

$$T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\psi & 0 & \sin\psi \\ 0 & 1 & 0 \\ -\sin\psi & 0 & \cos\psi \end{bmatrix}$$

Given $X = [0, 0, 1]$ and $X_0 = \tau$, it is easy to figure solve ϕ and ψ, thus get to know T. We then transform all the points in the local patch with rotation matrix T and get a calibrated point cloud.

2.5 Descriptor calculation



(a) A 3D model made up of two connected planes with an obtuse angle, which can be observed from two perspectives.

(b) Sampling result from view 1 after viewpoint calibration. Each dot in the figure stands for a pixel in captured RGB-D image.

(c) Sampling result from view 2 after viewpoint calibration. Each dot in the figure stands for a pixel in captured RGB-D image.

Figure 3: How viewpoint can affect sampling density.

With point cloud as input, there is a new problem emerged that the sampling density will be affected by viewpoint. As Fig. 3 shows, if we have a 3D model made up of two connected planes (with obtuse angle) and view it from two different perspectives, we will get different sampling results after camera view calibration. This is because the resolution of the sensor is the same among all the regions, therefore the surfaces facing the sensor will appear larger and get sampled finely, while the other one nearly along camera view will appear small in the image and get much less sample points. This density variation from viewpoint change not only brings noise into normal vector estimation, but also invalids classical descriptors based on point density/number statistics such as Shape Context [2].

Therefore, the descriptor should have three properties: 1) Resistant to density variation from viewpoint change; 2) (2D) orientation invariant; 3) compact.

Based on these expectations, we first project the texture of the point cloud to calibrated camera imaging plane. Considering not every pixel in the imaging plane will be filled, i.e. some regions in the projected image will not receive any pixels in the 3D patch, we use k-nearest neighbor interpolation to fill the areas lacking color information, as Fig. 4 shows. This step is mainly aimed to solve the density variation problem. After projection and interpolation, the resolution of the projected image is uniform again. And the image texture will not change much with only changing the viewpoint.

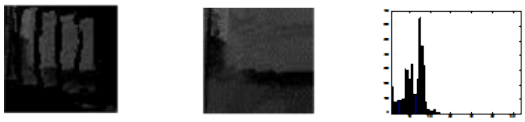


Figure 4: Procedures to compute descriptors. The left figure shows a sample image projected from a 3D local patch. Middle figure shows the result after kNN interpolation. And the right figure is the final descriptor.

To achieve orientation invariance and compactness, we adopt color histogram. The final descriptor is calculated as 64-bin color histogram of the projected patch and normalized according to L1 norm.

3. EXPERIMENTS AND DISCUSSIONS

To evaluate the effectiveness of our descriptor, we use two RGB-D images from different view of the same scene, as Fig. 5 shows to test. First two descriptors, SURF and Viewpoint Invariant descriptor (combined with both SURF detector) are calculated from two test images, and then we do nearest neighbor search among all the patches. In the end, 75 best matches are selected and visualized in Fig. 6.

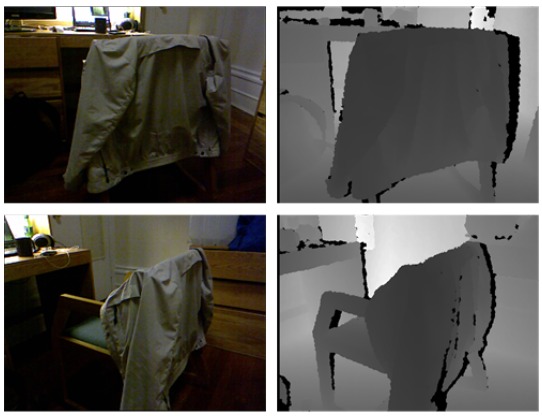


Figure 5: Input images for experiments, with RGB-D images of a chair from different views.

We can see our descriptor achieves better matching quality, especially in regions changed obviously with viewpoint change. However, some matching are not accurate, this is mainly because color histogram is sensitive to illumination variance.

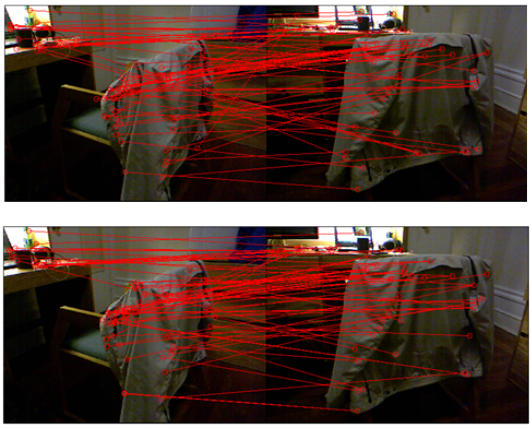


Figure 6: Experiment result with nearest neighbor matching. The image above shows matching result of SURF, and the image below shows result of our Viewpoint Invariant descriptor.

4. CONCLUSION AND FUTURE WORK

In this paper, we present a novel descriptor with viewpoint invariance based on RGB-D images from depth sensor. Combined with scale-invariant detectors, our descriptors can achieve 3D affine transform invariant, which is supported by some preliminary experiments. Currently there is still an unresolved issue, that if we simply use 2D scale invariant detector, it is common to get interest points on the boundary of two objects, where normal vector is ill-defined. Therefore, the future direction is two folds, one is trying to solve this problem by proposing a RGB-D image-based detector which extracts interest points directly from 3D model, the other is trying to improve matching performance of current descriptor, possibly with introducing 3D affine transform estimation based on RANSAC, employing more stable descriptors instead of color histogram, and adopting machine learning techniques such as mentioned in [4].

5. ACKNOWLEDGMENTS

The authors would like to thank Dr. Rongrong Ji for the helps in idea formulation.

6. REFERENCES

[1] H. Bay, T. Tuytelaars, and L. V. Gool. SURF : Speeded Up Robust Features. In *European Conference on Computer Vision (ECCV)*, pages 404–417, 2006.

[2] S. Belongie, J. Malik, and J. Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *In NIPS*, pages 831–837, 2000.

[3] T. Dey, G. Li, and J. Sun. Normal estimation for point clouds: a comparison study for a Voronoi based method. In *Proceedings Eurographics/IEEE VGTC Symposium Point-Based Graphics, 2005.*, pages 39–46, 2005.

[4] L. Ding and P. Zhao. Semi-supervised learning with varifold laplacians. *Neurocomput.*, 73:1580–1586, June 2010.

[5] J. Garcia and Z. Zalevsky. US patent 7433024: Range mapping using speckle decorrelation. http://ip.com/patent/US7433024.

[6] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov. 2004.

[7] Microsoft. Kinect. http://www.xbox.com/en-US/kinect.

[8] K. Mikolajczyk and C. Schmid. Performance evaluation of local descriptors. *IEEE transactions on pattern analysis and machine intelligence*, 27(10):1615–30, Oct. 2005.

[9] R. Ohbuchi. Salient local visual features for shape-based 3D model retrieval. *2008 IEEE International Conference on Shape Modeling and Applications*, pages 93–102, June 2008.

[10] P. Papadakis, I. Pratikakis, T. Theoharis, and S. Perantonis. PANORAMA: A 3D Shape Descriptor Based on Panoramic Views for Unsupervised 3D Object Retrieval. *International Journal of Computer Vision*, 89(2-3):177–192, Aug. 2009.

[11] M. Pauly, R. Keiser, L. P. Kobbelt, and M. Gross. Shape modeling with point-sampled geometry. In *ACM SIGGRAPH 2003 Papers*, SIGGRAPH '03, pages 641–650, New York, NY, USA, 2003.

[12] C. Wu, B. Clipp, X. Li, J.-m. Frahm, and M. Pollefeys. 3D Model Matching with Viewpoint-Invariant Patches. In *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2008.

[13] J. V. Wyngaerd, L. V. Gool, R. Koch, M. Proesmans, B. Leuven, E. T. H. Zentrum, and C. Zurich. Invariant-based registration of surface patches. In *The Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1999.