Money Measure
Measure Distances in Images Using a US Dollar Bill

Levi Daniel Franklin
Computer Science Department
Stanford University
Stanford, United States
lefrankl@stanford.edu

Abstract—The goal of this project is to make it easier to measure objects in day to day life. This is normally difficult to do because a ruler or tape measure is required, which is difficult to come by in many situations. Money Measure works by allowing you to use a smartphone camera to measure things. By taking a photo of an object next to a common US dollar bill, Money Measure can measure the distance between any two points in the scene. The dollar bill is automatically detected and utilized as a reference to real world distances. Additionally, the image is analyzed such that the image can be taken from a non-perpendicular angle the measurement will still be accurate.

I. INTRODUCTION

Cell phones today can be utilized for many things. As time has gone on cell phones have gone from devices used to place calls, to intricate computers that can take pictures, send texts, access the internet, play music, and much more. People find that they need to carry less and less supplies with them at all time in order to accomplish day to day tasks. However, one thing that many people do not have the ability to do in normal circumstances is measure distances. In order to measure something, a person needs to come by a ruler or a tape measure of some kind. This is often not feasible as they are not ubiquitous in the everyday world.

This project aims to provide a mechanism by which an everyday person can utilize their cell phone to measure short distances. To achieve this, a US dollar bill is used as a reference of real world distances. In computer vision it is possible to use various techniques to garner information about a 3d scene from an image such as parallel lines and vanishing points. However, the problem of rebuilding a scene using these techniques can only be solved “to scale.” Information about real world distances in an image is necessary to provide that scale. Because the size of a US dollar is rigidly defined, one can use it to recover some distances in an image. Further, because all dollar bills have roughly the same appearance they can be detected in an image. Using a dollar bill to detect distances makes sense as they are easily obtained by most everyone at all times. Money Measure provides a seamless, easy to use method to measure things using just your cell phone and a dollar.

II. PREVIOUS WORK

Lots of work has been done in the area of detecting objects in images. Many techniques have been developed that have various shortcomings for detecting an object. The simplest techniques involve taking a “template” of your object and sliding it over your image at every pixel. [3, 4] This, however, is computationally intensive and not very robust. Any deviations in illumination, scale, or rotation will not be detected. Possibly the most popular methods for detecting objects is the SIFT or SURF detectors. [1,2] Both of these methods involve using various techniques to find “keypoints” in an image. Keypoints are points in the image that are clearly identifiable such as corners. Keypoints can be found in a template of your object and matched with your image. One shortcoming of this technique is that it only works with one image. If you are trying to detect an entire class of objects, this technique will do little to help. Another method is to use a visual “bag of words”. This technique takes into account the special relation of various features in an image and also allows you to use several images to find a class of objects.

III. TECHNICAL SUMMARY

The method of object detection that is used in this project is a keypoint detector/descriptor similar to the popular SIFT and SURF techniques. [1,2] Because the SIFT and SURF techniques are patented, they are not included in the android version of opencv. However, another similar detection algorithm called ORB (Oriented FAST and Rotated BRIEF). [5] This algorithm works by using FAST [6] to find keypoints, and then BRIEF [7] to generate descriptors for those points. These descriptors allow points to be matched regardless of scale, illumination, or orientation.

In order to assist in the detection of keypoints, a method is used to help eliminate background pixels from keypoint consideration. A multidimensional MAP detector was created offline with a training set of several dollar bills. The pixels of these training images were placed into a 3 dimensional RGB space. Then this space was divided up into 16x16x16 classes. If a threshold was passed (500 pixels) in that region, the class is considered a dollar. All other classes are considered background. The threshold is chosen through testing somewhat liberally in order to have very few false negatives. This mask of pixels is applied to the image detection in order to not detect
points in the background. This allows us to use the color of the dollar to assist in its location.

The resulting binary image after MAP detection is very noisy. Many points within the dollar are marked as background despite being within the dollar confines. To combat this, a dilation of the image is done. This expands the “dollar” region to erase holes created by noise. Once this dilation takes place, the dollar is generally one large contiguous region, mostly free of “background pixels.”

Once they keypoints of a dollar are detected in an image and a template, they are matched using a brute force search. The 2 nearest neighbors are found and a ratio test is used to determine if a match is sufficiently good or not. Once good matches have been found, RANSAC is used to refine the matches further and generate a homography. This homography is then used to turn distances between pixels in the image, into real world distances.

IV. TECHNICAL DETAILS

A. ORB Detector/Descriptor

The ORB [5] detector is the method by which a dollar is located. This technique has two steps. The first step is to discover keypoints in an image. Keypoints are points in an image that can be considered “significant” and can be found again in a similar image. This includes image features like corners. Interestingly, points along edges do not make good keypoints as they are not easy to relocate.

Keypoint detection is handled using FAST [6] (Features from Accelerated Segment Test) detection. This is done by examining a circle of 16 pixels around each pixel. If the intensities of 12 contiguous pixels are all above or below the pixel intensity by a threshold, it determines a keypoint is found.

Keypoints that are found must also have a description associated with them. This allows features to be matched even when they are transformed by rotation, illumination, or scale. The method that is used to describe these keypoints is BRIEF [7] (Binary Robust Independent Elementary Features). BRIEF works to find a description of a keypoint that will allow the keypoint to be rediscovered even in new images. ORB works to make this recognition robust even to rotation by calculating a set of varying angle BRIEF descriptors.

B. Multidimensional MAP Detector

In order to take into account the color of a dollar bill in detecting it a MAP detector was created. This detector was trained on 5 images of dollar bills in varying lighting. Each pixel is taken and placed into a 3 dimensional histogram, 16 buckets for each color channel in RGB.

When a new image is processed, each pixel is passed through this 16x16x16 histogram and the resulting bucket is found. If the number of “dollar” pixels is higher than a threshold (500), this pixel is considered a dollar pixel.

C. Dilation and Erosion

After the MAP detection is complete the image is quite noisy. Many false negatives and false positives arise out of the MAP classification. To combat this a method of removing holes in a region is used called dilation. Dilation involves taking a structuring element and passing it over each pixel in an image. The structuring element used in our pass is a disk of radius 5 pixels. If any of the pixels under this disk are white, and therefore dollar pixels, the point in the middle is marked as white. This allows the white spaces to expand outward and close any holes in the image.

The opposite of this technique, an erosion, is then applied to return the image to its original state. Any holes that closed during dilation, however, remained closed. This leaves the white space as one continuous region.

D. Measuring Distances

Once ORB keypoints are found in both the template and image and have descriptors associated with them, a matching is done. The L2 distance between the descriptors of each template keypoint and the image keypoints is found. The 2 nearest neighbors based on this distance are returned and a ratio test is done to find out if they are a good match. As is suggested in the SIFT paper, a ratio of less than .8 between their distances suggests a good match.

Once the ratio test is applied, RANSAC is done to find a good match. Because the points are part of the same image, a simple homography transformation between the sets of points should apply to any points that are true matches. To find this homography, 4 random points are chosen repeatedly to make a test homography and the transformation is computed for all keypoints. The homography with the most keypoints matches that fall within a threshold is the correct one.

This homography is then used to transform coordinates in the template reference frame and the image reference frame. The 4 corners in the template image, multiplied by the homography, allow us to draw the corners of the dollar. Then, once the user selects two points in the image, the inverse homography allows us to transform them into the template coordinate frame. Because we know the size of the dollar in that frame, we can then compute the distance between those two selected pixels.

V. RESULTS

The finished app is tested by taking pictures of a dollar in a scene with an object whose distance is known, such as a ruler. After running the application and testing it using a ruler that was 100 mm long the measurement consistently fell within 2-3 mm of the correct length. (Figure 1). It was also found that the algorithm works fairly well for when the image is occluded. In one test, a laptop screen that is 15 inches wide was measured by holding the dollar against it. (Figure 2) The resulting measurement was 382.96 mm which is 15.07 inches. Overall the system seems to be fairly accurate.
VI. FUTURE WORK

There are many areas that this work could be applied in the future. One obvious way this work can be extended is by applying the algorithm to other objects. The first consideration is to allow the use of other US currency. This can further be extended by using foreign currencies. Another possible expansion is the use of coins.

A more ambitious extension might be to attempt to build a large database of objects that can be identified. Many common objects have rigidly defined sizes so these could be incorporated easily. A more difficult task would be to attempt to use objects that have a fairly common size, but which are not defined quite as rigidly, such as a pencil, or someone’s hand.

Another future pursuit is to attempt to find a dollar in real time. The algorithms that are employed in the dollar detection currently take several seconds to execute. Finding a dollar in a video feed would allow the user to have almost instant feedback on the distance between points. This could be extended even further by allowing a person to mark a point and then scan their camera over an area to measure larger distances.

ACKNOWLEDGMENT

I would like to thank the course director, Bernd Girod, as well as the course assistants, David Chen and Huizhong Chen for their assistance throughout the course.

REFERENCES

[4] Canny, John, A Computational Approach to Edge Detection, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139