INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 2)
Available online at: www.ijariit.com

3D reconstruction of regular objects from multiple 2D images using a reference object

Krishna Sai Joga
krishnasaijoga@gmail.com
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

K. Kavya Sree
kavyak0298@gmail.com
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

Navya Spandana
knavyaspandana.15.cse@anits.edu.in
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

G. Gowri Pushpa
gowripushpa.cse@anits.edu.in
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

ABSTRACT

The dimensional analysis of an object from an image reduces a lot of burden for a user, like the traditional measuring tape method. Using the dimensions will make reconstruction of the 3D model of the real-time object easier. However, this method is not used in the current implementation. Dimensional analysis can also be helpful in online shopping where the user's availability for fitting is not possible. 3D model replaces the fitting stage in online shopping. Once the dimensions of an object's surface are found, it is easy to calculate surface areas, given surface areas we can calculate volume. But the calculation of volume requires more than one dimension of the object. In this paper, an approach using a reference object, whose real-time dimensions are already known is used. The whole process is divided into three tasks - Object detection using SURF algorithm, Dimensional Analysis of the 2D object using pixel per metric ratio given that there is a reference object on the same plane and 3D reconstruction using Structure from Motion algorithm.

Keywords— *Object detection, SURF, Dimensional analysis, Pixel per metric, 3D reconstruction, Structure from motion*

1. INTRODUCTION

Computer vision [1] is a wide area that deals with image extraction, segmentation and 3D reconstruction[11]. The dimensional analysis involves the calculation of real-time dimensions of an object like length and breadth. The traditional method of calculating dimensions of an object is using a measuring tape, a scale, vernier calliper or any physical measuring device. These methods might not be feasible for all real-world applications. When dimensions of an object from a digital image are already known, calculation of surface area, volume, weight, density (if the mass is known) can be done easily. The evolution of smartphones has increased the availability of digital cameras, using them is much than measuring tapes. Dimensions of height and breadth can be calculated using a single 2D image, another view of the object is required to get the information necessary for depth calculation. However, 3D object reconstruction [11] requires multiple 2D images. The proposed model uses a reference object along with the main object in the image. It can be a matchbox, rupee coin, measuring scale, etc.

In this paper, a method is described for the calculation of dimensions of an object using a reference object on the same plane. It employs SURF [2] algorithm to detect the reference object, which can be one among matchbox, rupee coin, etc. whose real-time dimensions are already known. After knowing the reference object, by using canny edge detection we get edges of the required object. Using pixel per metric ratio we can find the dimensions of the required object in a 2D image. In order to find the dimensions of the required object, we place a reference object with it as mentioned above.

There are few limitations here, the size of the reference object is known to us and SURF [9] algorithm considers the leftmost object in the image as the reference object and maps the image with the keyholes and edges of the images in the dataset. After mapping the keyholes, the image is recognised and then comes dimensional analysis. Canny edge detection [13] is applied only after implementing gaussian blur to the 2D image. The Gaussian filter smoothens the image to reduce noise and unwanted details and textures. The Canny edge detector detects the edges of the objects in the image and leftover edges can be corrected by applying to dilate and erode technique. Then pixel_per_metric[12] method is applied to calculate the no. of pixels per metric, this determines the size of the objects in the 2D image.

2. SYSTEM ARCHITECTURE

The system implements two modules to produce the desired output. One being dimensional analysis, where the dimensions of the main object are calculated using the already known dimensions of the reference object. The next step is 3D Reconstruction, where the input images and object dimensions are used to reconstruct the 3D object.

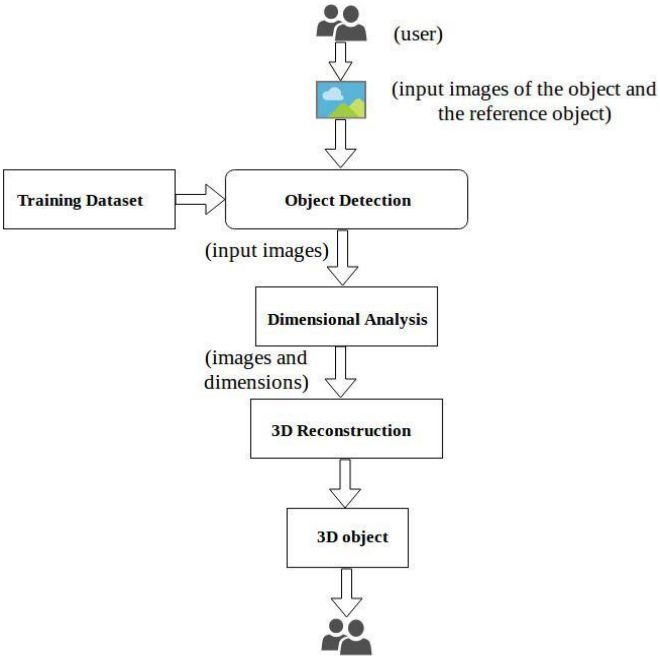

Fig. 1: 3D Reconstruction of System Architecture

3. PROPOSED METHOD

The proposed model performs three tasks - object recognition, dimensional analysis and 3D reconstruction. For object recognition, the input 2D image consists of the main object and a reference object lying on the same plane. The main object is restricted to be a regular one. The reference object can be a coin, matchbox, Rubix cube, etc. whose real-time dimensions are already known. SURF[9] algorithm is used for the recognition of both the main object and the reference object. Once the objects are recognized, using the real-time dimensions of the reference object we can calculate it's dimensioned in the image. Later, using the available information and pixels per metric ratio method we can calculate the dimensions of the main object from the image. Over the years, many projects have been developed for object recognition, dimensional analysis and 3D reconstructions individually. We plan on developing a single project that'll perform all these tasks.

- Object Recognition
- Dimensional Analysis
- 3D Reconstruction

3.1 Object Recognition

For object recognition, SURF[2] (Speeded-up Robust Features) algorithm is used. Prior to this algorithm SIFT[10] algorithm is used, but it was comparatively slow and people needed a speeded-up version. In 2006, Bay H., Tuytelaars T. and Van Gool L. introduced SURF[2]. For finding scale-space, in SIFT[10], Lower approximated Laplacian of Gaussian(LoG)[3] whereas SURF approximates LoG with a box filter. The advantage of such approximation is that convolution with the box can be easily calculated with the help of integral images. And it can be done in parallel for different scales. Also, SURF relies on the determinant of the Hessian matrix for both scale and location.

For feature description, SURF[2] uses wavelet responses in both vertical and horizontal directions. A neighbourhood of 20sX20s, where s is size, is taken around key point, which is further divided into 4X4 subregions. For each subregion, horizontal, vertical wavelet responses are taken and a vector of form is formed. This gives SURF[9] feature descriptor with 64 dimensions. For more distinctiveness, SURF[9] has an extended 128 dimension version.

$$v = \left(\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|\right) \tag{1}$$

3.2 Dimensional Analysis

To calculate dimensions of an object from an image there are two methods: Focal length method and pixel density method. Focal length method works on a principle of pinhole projection formula[4][5], which states that ratio of object length in terms of pixels to real-world dimensions of an object is equal to the ratio of focal length of the camera to distance between camera and object. The formula below:

$$x/f = X/D \tag{2}$$

In the above formula, we have a length of the object in terms of pixels(x), real-world length of the object(X), the focal length of a camera lens(f) and a distance between camera lens and object(D). This can be used to calculate real-world dimensions of the object. For this method, we need to know the distance between the camera and lens(D), which is an extra parameter for us.

$$Orb = \frac{Opb * Rrb}{Rpb}$$

$$Orl = \frac{Opl * Rrl}{Rpl} \quad (3)$$

Pixel Density method requires a reference object in the plane along with the object. The reference object is one whose real-world dimensions are known prior to us. The user is free to choose any object as a reference object gave it satisfies given conditions:

- The reference object is coplanar with the object in the image
- The reference object is easily identifiable in the image

To know the pixel lengths of the objects, lines are drawn around the edges of the objects in the image. Then using Euclidean distances from points we get pixel lengths. Using the real world length of the reference object, pixel lengths of the reference object, we get Conversion Factor(CF). CF is the ratio of object_width/known_width, where object_width is real-world dimensions of the object and known_width is real-world dimensions of the reference object. This CF is required to convert dimensions from the image(pixel length) to real-world dimensions.

Fig. 2: Edges drawn around the object for dimensional analysis

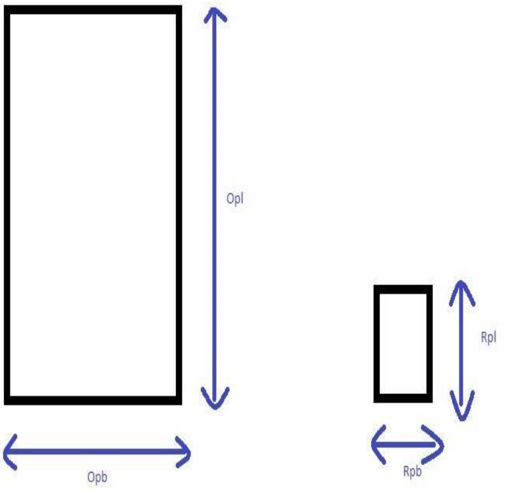

Fig. 3: Reference Object and Require Object

The proposed model detects objects that are trained with images which contain objects including the reference object. This reference object when detected we get its dimensions. This model is an automated approach where objects are detected as contours in the image and find its dimensions. The images are loaded from the hard drive of the computer, they are converted to grayscale and smoothened using 9X9 gaussian filter.

The Gaussian filter is highly effective to remove noise from the image. To create an edge map of all objects in an image, the Canny edge detector is used. Dilation and erosion are used to fill in any gaps generated in the edge map. In this method, a threshold of 100 is used, assuming that the object of interest occupies the majority of the area. An object occupying less than the threshold is automatically ignored as an edge. And ones with the area above threshold, bounding boxes are drawn around objects in the image. As CF ratio is already defined, we calculate the real world dimensions from the pixel lengths we get from boxes drawn.

3.3 3D Reconstruction

Photorealistic 3D models are used in a wide variety of applications like movies, games, through to simulation, training, reconstruction of historic monuments. In this model, we propose to use the Structure from Motion(SfM) technique. This technique uses multiple 2D images of a scene or an object to generate a point cloud-based 3D model similar to LiDAR technique. It results in a high definition 3D model using consumer grade cameras. SfM is based on stereophotogrammetry[], which uses triangulation to calculate relative 3D positions from stereo pairs. Given different overlapping images, key points or matching features are detected. These distinguishing features are usually corners or line segments.

These features are tracked from image to image to produce camera positions and orientations in 3D space, which is a point cloud. The minimum requirement for SfM to work is a minimum of three visible images. More the number of images SfM works better. For faster processing and better quality of the model, it is better to have the images taken at the same time with the same camera but this is not a requirement.

Scene Reconstruction

Step 1: Automatically estimate position, orientation and focal length of the camera
Step 2: Plot 3D position of feature points in 3D cloud space

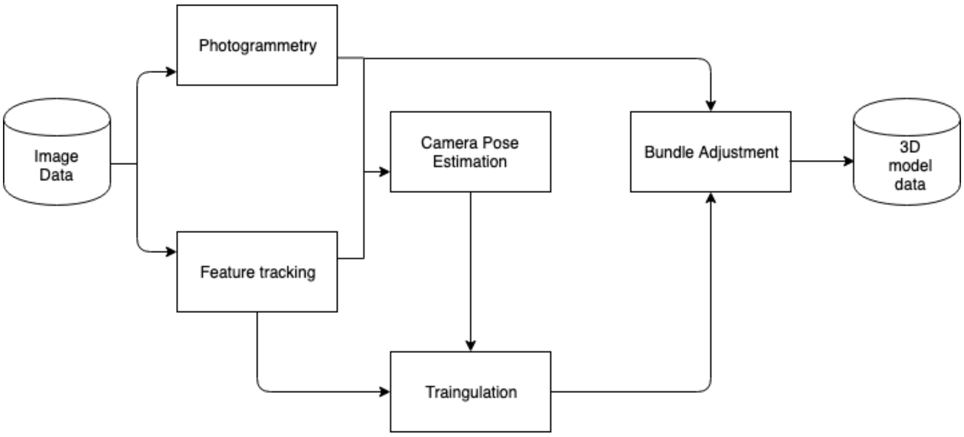

Fig. 4: Flow chart of 3D reconstruction

Bundle Adjustment: Structure-from-motion is the problem of recovering the 3D structure of the scene and the camera motion from a set of images. Bundle adjustment is a particular optimization algorithm used to solve it. When the cameras' intrinsic parameters and camera extrinsic parameters (i. e. camera poses) are known, you can actually compute the point cloud from the matching points using multi-view triangulation without bundle adjustment. You need to do nonlinear optimization when your estimate of the camera poses is uncertain, and bundle adjustment is the standard algorithm used for that. Multi-view structure from motion (SfM) estimates the position and orientation of pictures in a common 3D coordinate frame. When views are treated incrementally, this external calibration can be subject to drift, contrary to global methods that distribute residual errors evenly. Here the method proposes a new global calibration approach based on the fusion of relative motions between image pairs.

Preprocessing: In this paper, contraharmonic mean[14] based JPEG compression algorithm is gaged. The images compressed with the suggested approaches hold less encoded bits and as a result transmission rate is enhanced. The contraharmonic mean is higher in value than the arithmetic means and also higher than the root mean square. The name *contraharmonic* may be due to the fact that when taking the mean of only two variables, the contraharmonic mean is as high above the arithmetic mean as the arithmetic mean is above the harmonic mean (i.e., the arithmetic mean of the two variables is equal to the arithmetic mean of their harmonic and contraharmonic means).

4. EXPERIMENTAL RESULTS

We implemented a dimensional analysis on the input image in figure 5 and obtain the desired output in figure 6. The proposed system is tested extensively for accuracy and general behaviour using both real and synthetic datasets. The implementation is written in python, and results are obtained on a MacBook Pro with a 2.66 GHz Intel Core i7 processor and 4GB of RAM, running Mac OS X Mavericks 10.9.1.

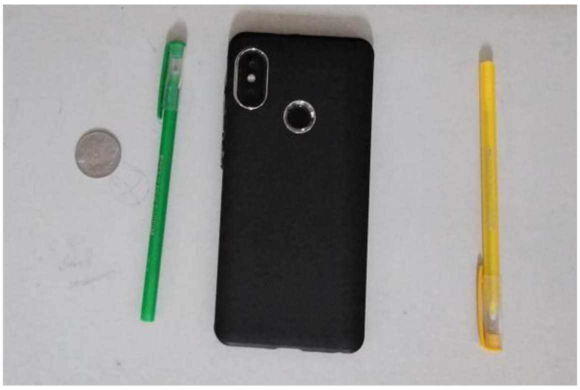

Fig. 5: Input image for dimensional analysis

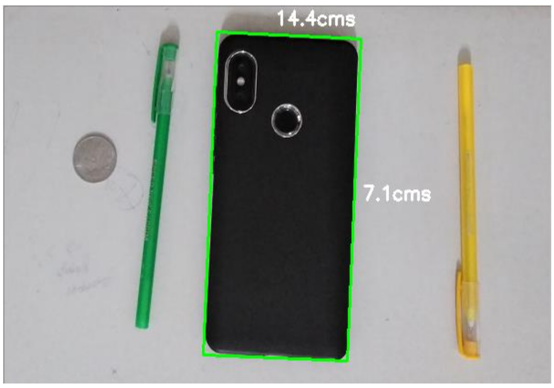

Fig. 6: Output of Dimensional analysis

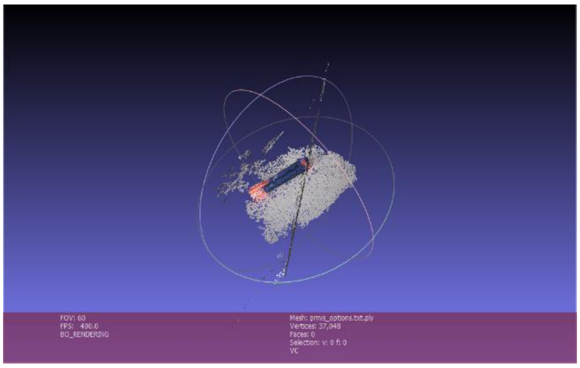

Fig. 7: 3D Model (1)

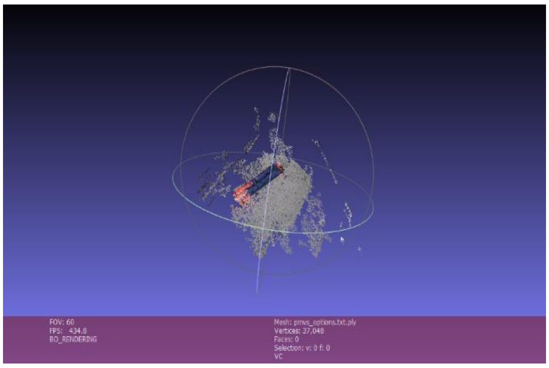

Fig. 8: 3D Model (2)

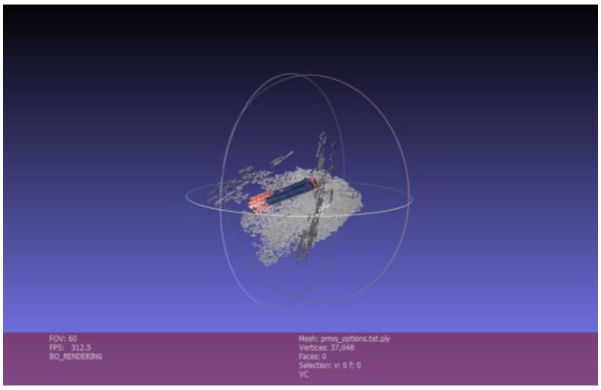

Fig. 9: 3D Model (3)

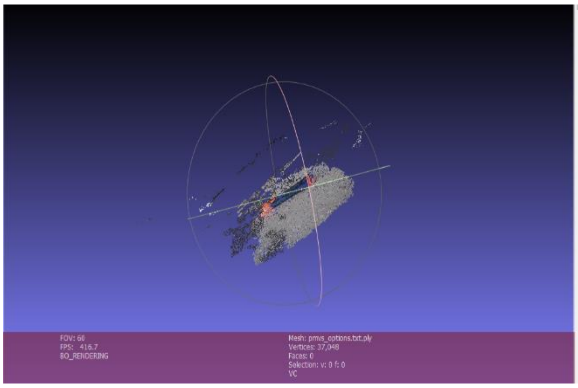

Fig. 10: 3D Model (4)

Table 1: 3D reconstruction results: Comparison between our method and classical method

Approach	Time Calculation
Our method	96
Classical method[6]	147

Table 1 represents the comparison between our method and the classical method [6] in function of calculation time and re-projection error. By using an incremental approach with the integration of a local bundle adjustment [7] that allows us to reduce the calculation time required for the global bundle adjustment [8] and also allows us to improve the 3D reconstruction quality.

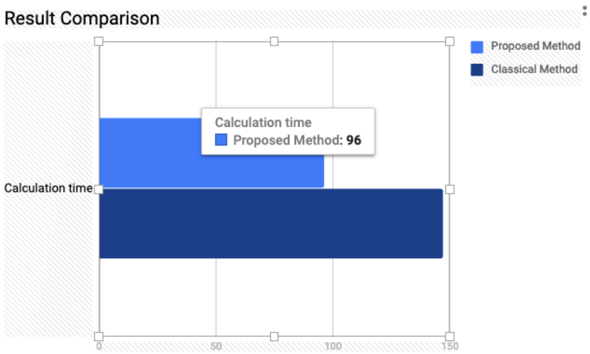

Fig. 11: Calculation Time of proposed method

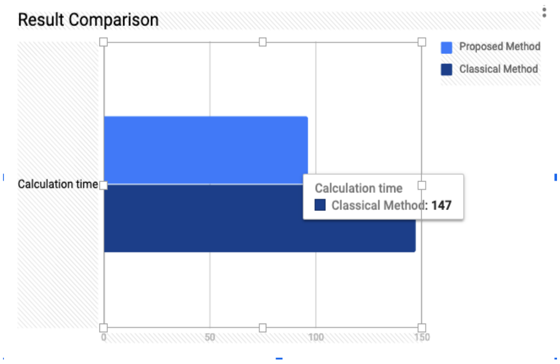

Fig. 12: Calculation Time of classical method

The above figures represent table 1 in the form of a bar graph which is a comparison between the time taken by the system to reconstruct the 3D model using a fixed number of 8 2D images.

5. CONCLUSION AND FUTURE WORK

In this paper, we discussed various modules like object detection, dimensional analysis and 3D reconstruction. There are various methods to implement these modules. The proposed methods gave very promising results. However, there is a lot of scope for extension and better accuracy. This paper presents a hybrid 3D reconstruction system that combines photogrammetry with SfM techniques. The system uses photogrammetric information to enhance the accuracy of SfM results. SfM provides a dense reconstruction, while photogrammetry is used to correct camera parameters or scene points that have high uncertainty. This procedure permits measurements between points that do not have corresponding targets, maintains the required metrological accuracy, and provides a more complete reconstruction than with either method alone. Results generated by the hybrid system for real and synthetic data demonstrate that both more accurate and denser reconstructions are obtained than with SfM alone.

The development of hybrid systems for 3D reconstruction helps improve both photogrammetry and structure-from-motion. The continued exploration of combining techniques in these fields may lead to both the improvement of current algorithms and the development of new ones. Investigation of photogrammetry-assisted volume-based reconstruction is an interesting and important topic for future applications. Moreover, additional visualization techniques are needed to show reconstruction accuracy, and to highlight key locations at which to place photogrammetry targets so that overall reconstruction accuracy is improved.

6. REFERENCES

[1] https://en.wikipedia.org/wiki/Computer_vision
[2] Herbert Bay, Tinne Tuytelaars and Luc Van Gool, SURF: Speed-up Robust Features,
[3] Brice, 2012, How to get the real-life size of an object from an image, when not knowing the distance between the object and the camera, StackOverflow, Available at http://stackoverflow.com/questions/9940140/how-to-get-the-real-lifesize-of-an-object-from-an-image-when-not-knowing-the-d, Cambridge, UK.

[4] Kayim, 2014, Calculation of Focal Length from Images, StackOverflow, Available at http://stackoverflow.com/questions/18232145/calculationof-focal-length-from-images, Cambridge, UK
[5] https://sci-hub.tw/https://ascelibrary.org/doi/pdf/10.1061/9780784479087.225
[6] https://sci-hub.tw/https://ieeexplore.ieee.org/document/7483225
[7] S. El hazzat, A. Saaidi, A. Karam, and K. Satori. Incremental Multi-view 3D Reconstruction Starting from Two Images Taken by a Stereo Pair of Cameras. 3D Research, volume 6, Issue Number 1, Article 11, 2015.
[8] M. Pollefeys, L. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch. Visual modelling with a hand-held camera. International Journal of Computer Vision, 59(3):207–232, 2004.
[9] SURF Algorithm for object detection (module 1) https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_surf_intro/py_surf_intro.html
[10] David G. Lowe, Distinctive Image features from Scale-invariant Keypoints, January 5, 2004
[11] https://en.wikipedia.org/wiki/3D_reconstruction_from_multiple_images
[12] https://www.pyimagesearch.com/2016/03/28/measuring-size-of-objects-in-an-image-with-opencv/
[13] https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_imgproc/py_canny/py_canny.html
[14] https://en.wikipedia.org/wiki/Contraharmonic_mean

APPENDIX

SURF - Speeded Up Robust Features
LiDAR - Laser Imaging, Detection And Ranging
SfM - Structure from Motion