6D Pose Estimation for Texture-less Objects

Harshal Soni
University of Alberta
hsoni@ualberta.ca

Navya Rao
University of Alberta
gururajr@ualberta.ca

Maryam Sedghi
University of Alberta
msedghi@ualberta.ca

Abstract

The developments in the field of Artificial Intelligence has instigated extensive research in computer vision to enable detection of objects adhering the autonomy and accuracy. In this work, we propose a technique to estimate the 6D poses of the Texture-less objects using deep neural network. The network is trained on T-Less dataset to estimate 3D transnational and 3D rotational pose estimates. However, fundamental properties of the objects, environment encompassing the object, and media to capture them poses a challenge to estimate the pose of the object; our work also includes a pose refinement technique in which our network iteratively refine the estimates by taking cosine similarity between observed image and the rendered image. The code for this work is available here : https://github.com/harshalsonioo1/MM811.git

1. Introduction

The dawn of object detection algorithms has kindled various applications in machine vision and industrial domains. For instance, in the domain of robotic manipulation, the ability to recognize the 6D pose of objects, i.e., 3D location and 3D orientation of objects, renders knowledge for motion planning in order to grasp the objects. Likewise, In a virtual reality or Augmented reality application, 6D object pose estimation enables the interaction between motion and objects. [11]

However, the sheer latency and cost to apply these algorithms to detect the transnational and rotational positions of the objects with varying geometry and textures have instigated the need for accurate pose detection techniques functioning efficiently in a cluttered, occluded scenario with a reasonable cost. Whilst the usage of depth component for object pose estimation is a viable option, such cameras have limitations with respect to frame rate, field of view, resolution, and depth range, making it very difficult to detect small, thin, transparent, or fast moving objects.

Albeit, RGB-only 6D object pose estimation is lucrative, it has some of the inherent challenges, viz, lighting, pose variations, and occlusions between objects. Furthermore, a robust 6D pose estimation method needs to handle both textured and texture-less objects.

Traditionally, the 6D pose estimation problem has been approached by co-relating the local features extracted from an image to features in a 3D model of the object [4, 21, 10]. They use the 2D-3D correspondences to recover the 6D pose of the object. Unfortunately, these methods relies heavily on few local features of the object and fail miserably for texture-less objects due to their geometry.

Detailed literature review delineates two classes of approaches used for texture-less objects. In the first class, methods learns to estimate the local key pints of 3D model coordinates from the input image using 2D-3D correspondence [23, 19, 29]. In the second class, 6d object pose estimation is seen as a regression problem [11]. These methods can tackle the surface dilemma of texture-less objects but the estimates are not accurate enough to make the method robust. To ameliorate the accuracy, certain pose refinement techniques based on hand-crafted image features [27] or matching score functions [19] are used.

In this work, we are employing two stage technique for 6D object pose estimation and refinement. In the first stage, a Pixel-wise Voting Network (PVNet) is used to produce an initial pose estimate of the test image. PVNet regress pixel-wise unit vectors pointing to the keypoints and use these vectors to vote for keypoint locations using RANSAC. This creates a flexible representation for localizing occluded or truncated keypoints and the provided uncertainties of keypoint locations can be further leveraged by the PnP solver. [17]

In the second stage, the refinement technique predicts a relative SE(3) transformation that matches a synthetic mesh rendered view of the object against the observed image. Then a new pose is computed to increase the matching score. This process is re-rendered in order to improve the pose estimates of the object.

This work makes the following contributions.
i) We introduced a pipeline for RGB based pose estimation-refinement that works efficiently for occluded objects effectively.

ii) Our iterative pipeline does not rely on any hand-crafted image features.
iv) We have conducted extensive experiments on the T-Less [9] datasets to evaluate the accuracy of our method. The rest of the report is organized as follows. After reviewing related works in Section 2, we describe our approach for pose matching in Section 3. Experiments are presented in Section 4, and Section 5 concludes the paper.

2. Related Work

There has been an eclectic mix of approaches devised to solve the above stated problem. The modern approaches can be classified into three categories:

2.1. Pose estimation from RGB images

Classically, matching local geometric features using RGB images is used [4, 21]. Under the gamut of this approach, a 3D model of an object is first extrapolated and local features are marked to it.Then, features extracted from the test image are matched to the features on 3D model and incorrect matches are screened out using robust estimation techniques such as [24]. Keypoint-based features such as SIFT [4] or SURF [1] are widely used. The 6D pose of the object can be recovered using the 2D-to-3D correspondences between the local features. This method can handle partial occlusion but these methods cannot handle textureless objects well, since rich surface measures is the prime necessity to detect these features robustly.

On the contrary, template-matching based methods are capable of handling texture-less objects [25, 14, 8]. Under the gamut of this approach, construction of the templates are constructed prior to training, where examples of templates are renderings of the object from the 3D object model or Histogram of Oriented Gradients(HOG) [5] templates from different viewpoints. Then during the testing phase, these templates are matched against the input image to estimate the orientation and distance of the test object in the input RGB. These approaches are highly restrained in their accuracy when objects are occluded.

Modern approaches employs deep learning to detect object key-points for matching or learn better feature representations for pose estimation [23, 12]. The state-of-the-art methods [20, 20] augment segmentation methods or deep learning [19, **?**, 11] based object detection for 6D pose estimation. Current state-of-art technique [26] uses an autoencoder to map the object in the image to a vector in a pregenerated codebook and simply index-searches for pose estimation.

Overall, deep learning based methods outranks traditional methods because of their efficient learning process.

2.2. Pose estimation from depth data

The second set of approaches employs the usage of depth images for 6D pose estimation. These set of approaches falling under geometric registration problem functions as follows: Given a 3D model of an object and an input depth image, the problem is formulated as aligning the two point clouds computed from the 3D model and the depth image,respectively. Speaking with a cursory glance, these methods can be divulged into local refinement methods and global registration methods. Iterative Closest Point (ICP) algorithm [2]and its variants [22]assumes an initial posture of the object and iteratively refine them to convergence.

Global registration methods [23, 15, **?**] utilizes iterative model fitting frameworks (RANSAC). In order to improve the registration performance, features on point clouds are also introduced for matching . A recent approach [28] propose to learn point features for registration,such as applying deep neural networks to point clouds [18].

2.3. Pose estimation from RGB-D

The dataset that comprises a RGB as well as depth component of the object sprung an approach that amalgamates these two for a better pose estimation. The most popular technique in this category [24, 7] obtains initial pose estimation using templates generated from 3D model of the objects, and then matching them against the input image. Afterwards, ICP refinement is applied on the initial pose estimates to refine them. [3, 7] regress each pixel on the object in the input image to the 3D coordinate of that pixel on the 3D model and applies set of correspondence using depth data to 3D model points. Final pose estimates is obtained by solving a least-squares problem. PoseCNN [30] is a popular framework that employs the aforementioned technique based on a neural network that amalgamates RGB images and depth images for3D translation and 3D rotational pose estimation, and an iterative pose refinement network using point clouds as input.

3. Framework

6D Object pose estimation techniques using RGB-only have made progress in recent years but they are still behind the RGB-D techniques. Our approach uses a pipeline that uses state-of-art pose estimation technique PVNet as its backbone and DeepIM iterative pose estimation techniques using RGB- only for final estimates. Additionally, pipeline is flexible to take advantage of other pose refinement algorithm such as ICP.

3.1. Dataset

As a part of this project, we are using T-less data set [9]. It has 30 industrial shaped object with no definitive textures captured using three synchronized sensors namely, Prime-

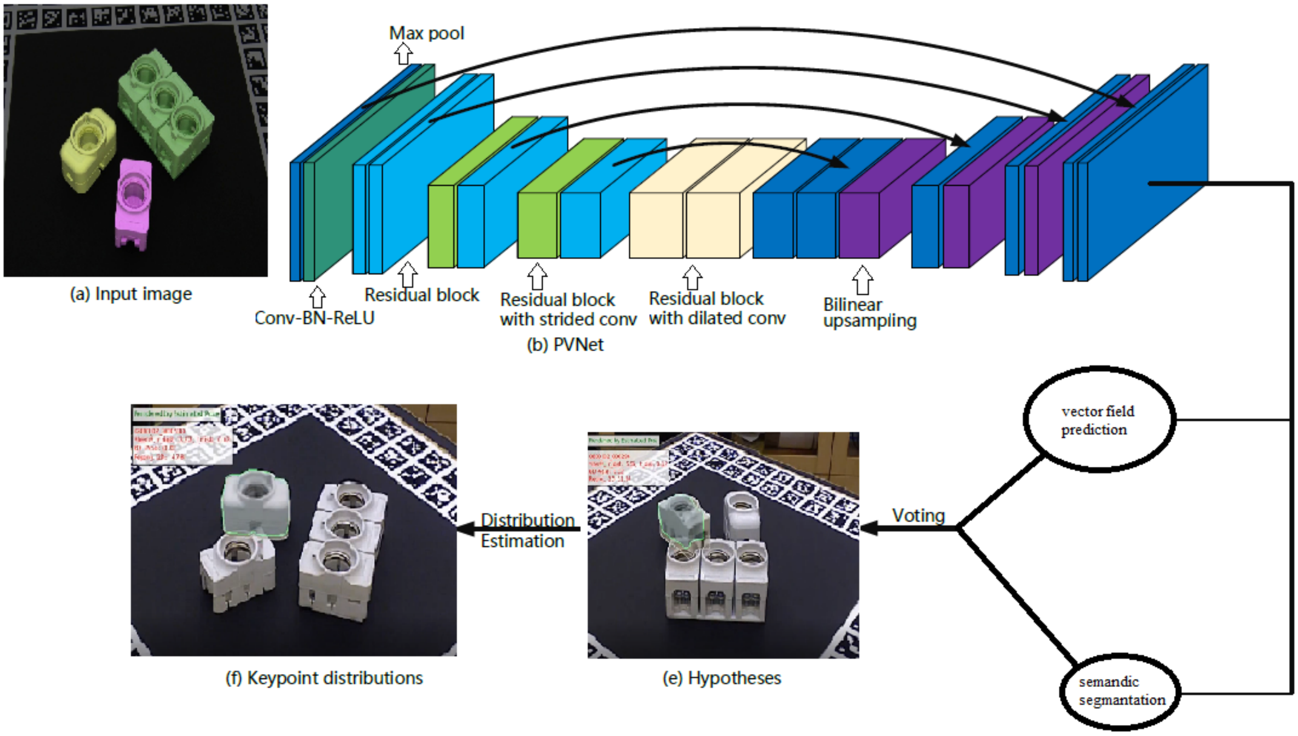

Figure 1. Architecture of Pixel-wise voting network

sense CARMINE 1.09 (a structured-light RGB-D sensor), Microsoft Kinect v2 (a time-of-flight RGB-D sensor), and Canon IXUS 950 IS (a high-resolution RGB camera).

Training images comprises of 38K footage from each sensor and testing images are numbered 10K from each sensor.

3.2. Initial Pose estimation

We estimate the initial object pose using Pixel-wise Voting Network(PVNet) as a backbone into two steps. First, it detects 2D object keypoints using RANSAC. This voting analogy provides a spatial probability distribution of each keypoint, facilitating us in the second step to estimate the 6D pose with an uncertainty-driven PnP algorithm. [17]

3.2.1 Keypoint Localization

Figure 1 overviews the first step of keypoint localization. PVNet predicts unit vectors representing every direction from object to each key point and pixel-wise object labels from a given image. This helps to extrapolate the hypothesis of 2D locations for any particular hypothesis and RANSAC-based voting is performed. As a derivative of these hypotheses, the mean of each keypoint is calculated and covariance with the respective mean is obtained.

3.2.2 PnP Solver

After deriving 2D keypoint locations for each object from the previous step, PnP problem solver namely, EPnP is used to compute 6D pose estimates. Also, reprojection errors are minimized to tackle the uncertainties.

3.3. Pose Refinement

We have used Deep iterative refinement(DeepIM) for pose refinement trained on T-Less dataset to directly output a relative SE(3) transformation that can be applied to the initial pose to improve the estimate.

As seen in Figure 2, initial 6D pose estimates are obtained from PVNet(pose(0)in the figure) and the 3D model of the object is used to generate a rendered image showing the mash representation of the object. This is provided as an input to the network which outputs an intermediate Δ pose. Now, this intermediate pose is iterated back into the system (pose(1)in the figure) along with the 3D model of the object until final refined 6D poses are generated as the network converges over a pre-defined threshold.

Figure 3. describes the mash renderer used for this purpose.

4. Experiments

We conduct extensive experiments based on T-Less dataset to evaluate our framework for 6D pose estimation

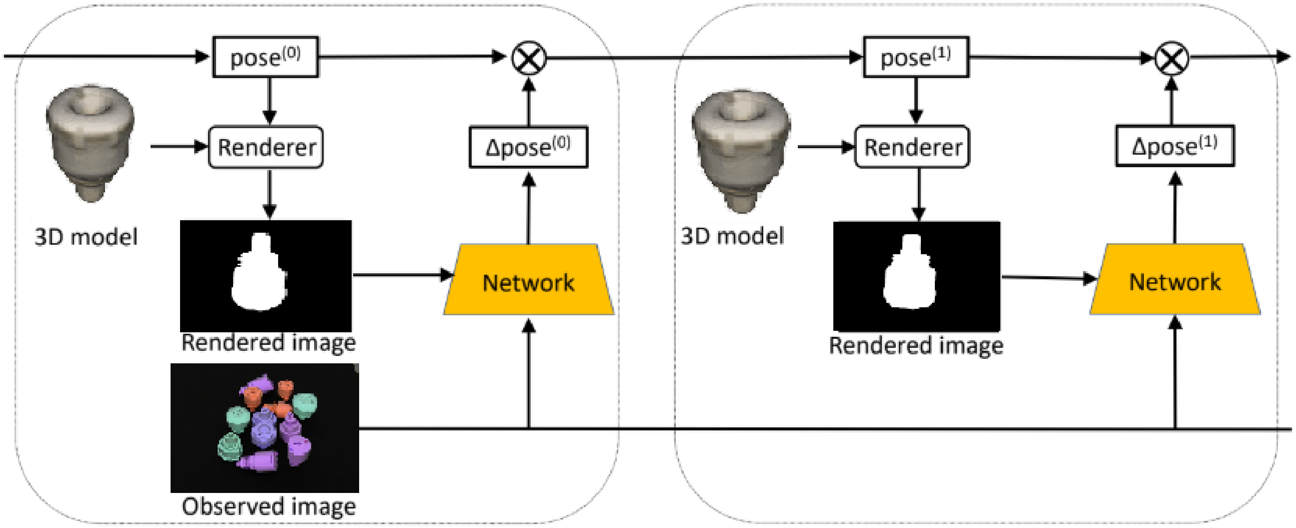

Figure 2. Iterative Pose refinement

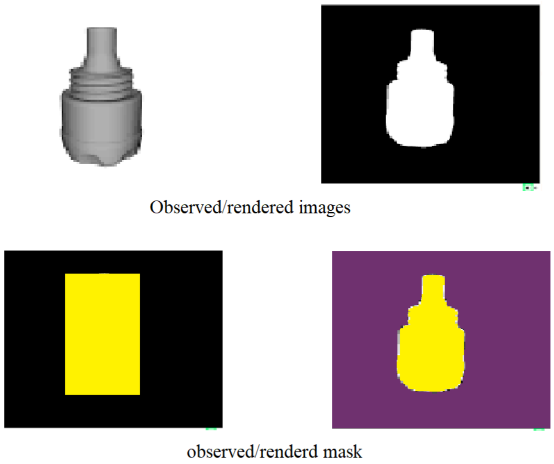

Figure 3. Illustration of rendered mask and original image used in refinement step

of objects. We also show that our network performs reasonably well for unseen objects.

4.1. Training Implementation details

1. Synthetic training data: Real training images are highly correlated and lacks occlusion. Therefore, we have used PASCAL VOC2012 dataset mixed ith our dataset to immune our network from overfitting.

2. PVNet: Inspired by [17], ”assuming there are C classes of objects and K keypoints for each class, PVNet takes as input the H*W*3 image, processes it with a fully convolutional architecture, and outputs the H*W*(K*2*C)tensor representing unit vectors and H*W*(C+1)tensor representing class probabilities”. We have used an MXNET based ResNet-18 for as our backbone. Furthermore,hypothesis generation, pixel-wise voting and density estimation using CUDA. The EPnP implementation is performed in OpenCV.

3. DeepIM: Inspired by [13], we have used the predefined network to train T-less dataset(images annotated with ground truth 6D object poses).Network based on FlowNet-Simple [6] is trained to learn the optical loss and foreground mask of the object in order to predict the relative transformation between the initial pose and the target pose using cosine similarity.

We have used Nvidia Tesla K80 graphics card for training purpose and with each training batch consisting 16 images. We have set the initial learning rate as 0.001 and halve it every 20epochs. 3 Objects from Primsense sensor and 3 from Kinect sensor are trained for 100 epochs.

4.2. Testing implementation details

Given a sample test image, we use the initial pose estimates from our PVNet implementation as the initial poses. Our DeepIM network runs at 25 fps per object using an NVIDIA 2080 Ti GPU with 2 iterations during testing to produce final 6D pose estimates.

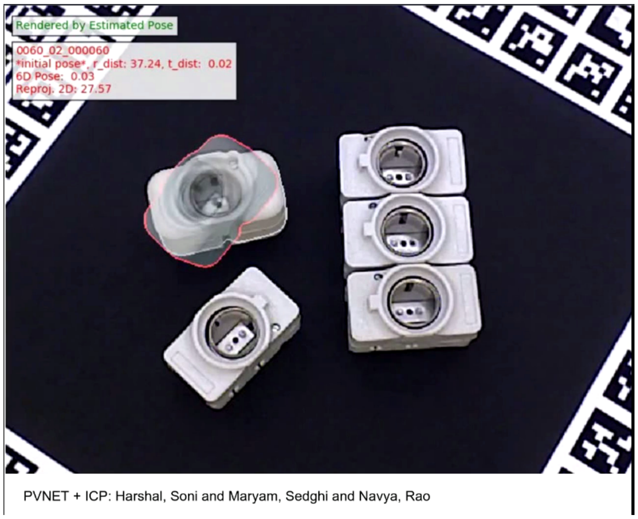
Rendered by Estimated Pose
0060_02_000060
initial pose, r_dist: 37.24, t_dist: 0.02
6D Pose: 0.03
Reproj. 2D: 27.57
PVNET + ICP: Harshal, Soni and Maryam, Sedghi and Navya, Rao

Figure 4. Figure 4. demonstrates an initial pose estimate

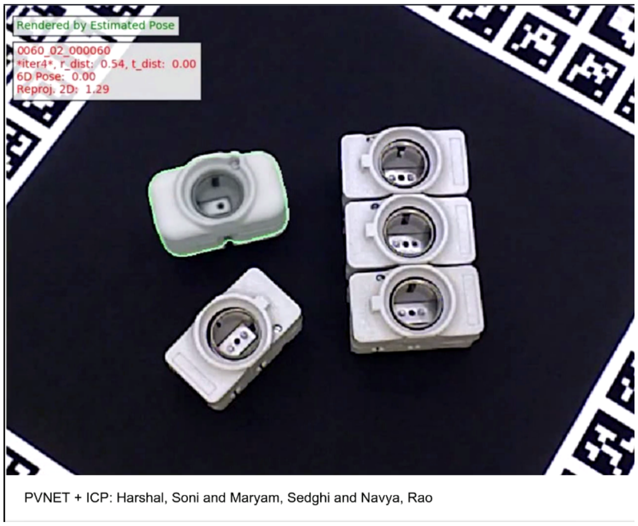
Rendered by Estimated Pose
0060_02_000060
iter4, r_dist: 0.54, t_dist: 0.00
6D Pose: 0.00
Reproj. 2D: 1.29
PVNET + ICP: Harshal, Soni and Maryam, Sedghi and Navya, Rao

Figure 5. Figure 5. demonstrates final pose estimate after refinements

4.3. Evaluation metrics

We have evaluated our method using ADD metric [16] and report AUC(VSD).

Plotting the Area Under curve for Visible surface detection for the objects belonging to Primesense sensor and Kinect sensor, we get the numbers which are approximately similar to the state-of-art method for 6D object pose detection, i.e, 0.875 and 0.904 respectively.

Autoencoder(2019)	Ours
0.890	0.875

Table 1. Results- AUC: Primesense Sensor

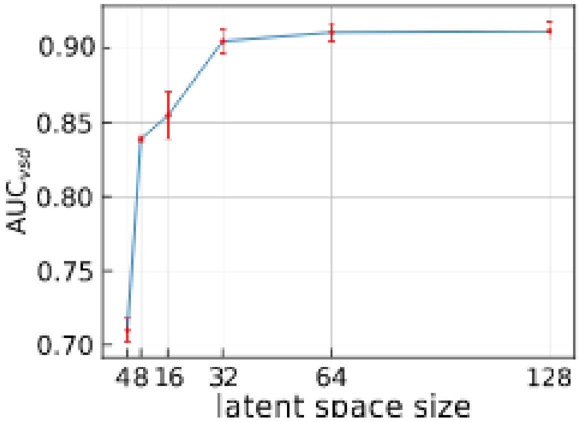

Figure 6. Effect of latent space size, standard deviation in red

Autoencoder(2019)	Ours
0.917	0.904

Table 2. Results- AUC: Kinect Sensor

5. Conclusion

In this work, we have demonstrated a simple and effective approach to estimate 6D object pose with respect to camera cardinal system. We have used ”Transfer learning” to encapsulate the state-of-art method for object pose estimation ”PVNet” for pose estimation and ”DeepIM” for iterative pose refinement into a single pipeline.The dataset employed for this project is “T-Less dataset” [29] and it has 30 industry-relevant objects with no discriminatory color, no texture, often similar in shape, some objects are parts of others.

A notable advantage of using this pipeline is parallel training of two networks for same object where the first network learns the 3D model of the object whilst the second learn the image profile of the same. Additionally, the performance of our method with almost marginally lesser than current state-of-art method for 6D object pose estimation due to dearth of training resources and time. Our efforts clearly indicates that in case of texture-less objects, pose refinement steps after an initial estimation is required because of their aberrant symmetric structures that makes key point detection difficult by conventional methods.

References

[1] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008. 2

[2] P. J. Besl and N. D. McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. International Society for Optics and Photonics, 1992. 2

[3] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother. Learning 6d object pose estimation using 3d object coordinates. In *European conference on computer vision*, pages 536–551. Springer, 2014. 2

[4] R. R. Churchill and A. V. Lowe. *The law of the sea*. Manchester University Press, 1999. 1, 2

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. 2005. 2

[6] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015. 4

[7] D. Frau-Meigs, I. Velez, and J. F. Michel. *Public policies in media and information literacy in Europe: cross-country comparisons*. Taylor & Francis, 2017. 2

[8] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian conference on computer vision*, pages 548–562. Springer, 2012. 2

[9] T. Hodan, P. Haluza, Š. Obdržálek, J. Matas, M. Lourakis, and X. Zabulis. T-less: An rgb-d dataset for 6d pose estimation of texture-less objects. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 880–888. IEEE, 2017. 2

[10] A. Kaiser, C. Holden, J. Beavan, D. Beetham, R. Benites, A. Celentano, D. Collett, J. Cousins, M. Cubrinovski, G. Dellow, et al. The mw 6.2 christchurch earthquake of february 2011: preliminary report. *New Zealand journal of geology and geophysics*, 55(1):67–90, 2012. 1

[11] W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1521–1529, 2017. 1, 2

[12] H. Krause. Krull–schmidt categories and projective covers. *Expositiones Mathematicae*, 33(4):535–549, 2015. 2

[13] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox. Deepim: Deep iterative matching for 6d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 683–698, 2018. 4

[14] B. Liu et al. Sentiment analysis and subjectivity. *Handbook of natural language processing*, 2(2010):627–666, 2010. 2

[15] N. Mellado, D. Aiger, and N. J. Mitra. Super 4pcs fast global pointcloud registration via smart indexing. In *Computer Graphics Forum*, volume 33, pages 205–215. Wiley Online Library, 2014. 2

[16] P. Moulon, P. Monasse, and R. Marlet. Global fusion of relative motions for robust, accurate and scalable structure from motion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3248–3255, 2013. 5

[17] S. Peng, Y. Liu, Q. Huang, X. Zhou, and H. Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4561–4570, 2019. 1, 3, 4

[18] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017. 2

[19] M. Rad and V. Lepetit. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3828–3836, 2017. 1, 2

[20] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 2

[21] F. Rothganger, S. Lazebnik, C. Schmid, and J. Ponce. 3d object modeling and recognition using local affine-invariant image descriptors and multi-view spatial constraints. *International Journal of Computer Vision*, 66(3):231–259, 2006. 1, 2

[22] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *3dim*, volume 1, pages 145–152, 2001. 2

[23] T. Sanna, H.-C. Diener, R. S. Passman, V. Di Lazzaro, R. A. Bernstein, C. A. Morillo, M. M. Rymer, V. Thijs, T. Rogers, F. Beckers, et al. Cryptogenic stroke and underlying atrial fibrillation. *New England Journal of Medicine*, 370(26):2478–2486, 2014. 1, 2

[24] H. Soltani, H. Taghirad, and A. N. Ravari. Stereo-based visual navigation of mobile robots in unknown environments. In *20th Iranian Conference on Electrical Engineering (ICEE2012)*, pages 946–951. IEEE, 2012. 2

[25] P. Sturm, S. Ramalingam, J.-P. Tardif, S. Gasparini, J. Barreto, et al. Camera models and fundamental concepts used in geometric computer vision. *Foundations and Trends® in Computer Graphics and Vision*, 6(1–2):1–183, 2011. 2

[26] M. Sundermeyer, Z.-C. Marton, M. Durner, M. Brucker, and R. Triebel. Implicit 3d orientation learning for 6d object detection from rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 699–715, 2018. 2

[27] H. Tjaden, U. Schwanecke, and E. Schomer. Real-time monocular pose estimation of 3d objects using temporally consistent local color histograms. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 124–132, 2017. 1

[28] J. Wang and X. Wang. *Structural equation modeling: Applications using Mplus*. John Wiley & Sons, 2019. 2

[29] R. Wright, E. Tekin, V. Topalli, C. McClellan, T. Dickinson, and R. Rosenfeld. Less cash, less crime: Evidence from the electronic benefit transfer program. *The Journal of Law and Economics*, 60(2):361–383, 2017. 1, 5

[30] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017. 2