Published: 23 August 2020

Citation in BibTeX format

ICCCV'20: 2020 the 3rd International Conference on Control and Computer Vision
August 23 - 25, 2020
Macau, China

RESEARCH-ARTICLE

Camera Pose Estimation using CNN

BHATTARABHORN WATTANACHEEP, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand

ORACHAT CHITSOBHUK, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand

Camera Pose Estimation using CNN

BHATTARABHORN WATTANACHEEP*
Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand
nroskool2@gmail.com

ORACHAT CHITSOBHUK
Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand
orachat.ch@kmitl.ac.th

ABSTRACT

Estimating camera pose is a significant process, which assures the success of the 3D modeling performance. This research presents a camera pose estimation using convolutional neural network (CNN) to transfer learning from pre-trained deep learning VGG19 model in order to extract features from a single image using several datasets captured in indoor and outdoor environments with diverse perspectives and photographic styles. Due to the large dimensions of the extracted features, Latent Semantic Analysis (LSA) are introduced prior to the CNN input. Then, the CNN is trained to predict the camera views and translations. The prediction performance is measured in terms of average mean square errors and compared to the reference techniques. As a result, the regression estimation of the proposed CNN model outperforms the others with average 0.24 degrees rotation error and 0.26 m. translation errors.

CCS CONCEPTS

• **Artificial Intelligence**; • **Computer vision**; • **Image and video acquisition**; • **Camera calibration**;

KEYWORDS

3D Reconstruction, Image Processing, Robotics, Deep Learning

ACM Reference Format:
BHATTARABHORN WATTANACHEEP and ORACHAT CHITSOBHUK. 2020. Camera Pose Estimation using CNN. In *2020 the 3rd International Conference on Control and Computer Vision (ICCCV'20), August 23–25, 2020, Macau, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3425577.3425593

1 INTRODUCTION

Nowadays, 3D modeling techniques are widely adopted in a variety of fields, such as robotics, aircraft, Unmanned Aerial Vehicles (UAVs) [1–3], navigating autonomous vehicles [4], mobile robotics and augmented reality [5], virtual application simulation and various components of large-scale localization, etc. The basic techniques commonly used in finding structures for 3D modeling are the Structure from Motion (SfM) and SLAM [6]. These techniques require feature extraction from two-dimensional images such as finding the angle of the object within the image and approximating geometric structure and camera poses to help estimate 3D models structural parameters. The process of finding image consistency depends on the ability of the feature detection technique to discovery the corresponding feature points, such as the SURF [7], ORB [8] or SIFT [9] methods. For the detection of feature points and image matching, the method requires an iterative comparison of the initial images in order to find rotation and translation within each pair of images. The size of the image dataset therefore affects the prediction time. RANSAC is usually used to aid the camera pose estimation in choosing pairs of feature points. If the pairing found in the first step is correct, the camera pose will be estimated successfully. Consequently, the limitations of feature matching in case of motion, blurring or light variation and the errors resulting from the different order of image matching pairs may lead to the failure in the above step. Moreover, if the estimation of the feature points is insufficient or their positions are errors, this makes it incapable of finding sufficient corresponding features between pairs of images resulted to predict rotation and translation.

Recently, the Convolutional Neural Networks (CNN) have been wildly adopted in related tasks such as image classification [10, 11], pattern recognition, image enhancement, object detection [12, 13], and semantic segmentation [14, 15] due to its high precision. In addition, the use of information transfer principles from these previously trained neural networks to transfer knowledge to other works. Since conventional machine learning and deep learning algorithms, so far, have generally been designed to operate in isolation, these algorithms are being trained to solve different tasks. When the feature-space distribution changes, the models must be reconstructed from scratch. Especially, when considering the context of learning for solving complex problems, most models require a large amount of information to learn. This leads to the difficulty in labeling the answers for each image in such a large training dataset. For example, ImageNet's dataset training requires more than a million images divided into different categories. Transfer learning is the concept to resolve the isolated learning model and derive the knowledge obtained for one task to solve relevant one. Therefore, a research, which transfers learning from CNN, starts with learning the desired job and modifies CNNs to estimate the camera position from the input image and extract reasonable features that are robust against motion blur and illumination for localization problems. PoseNet research [16] reveals that it is inappropriate to estimate camera poses from the high dimensional output of FC layers, since they cannot provide the best results due to the overfit problem from the PoseNet training data. From the above-mentioned PoseNet problem, subsequent research [17] introduced a modification of the PoseNet architecture using the GoogleNet architecture [12], deep learning network in which the softmax layer (for classification) was replaced by the FC layer. By [17] removing features of the FC layer,

*Place the footnote text for the author (if applicable) here.

ICCCV'20, August 23–25, 2020, Macau, China

ACM ISBN 978-1-4503-8802-3/20/08…$15.00
https://doi.org/10.1145/3425577.3425593

the authors then reshapes the feature vector into a 32 x 64 matrix for LSTM gesture estimation to minimize image encoding dimensions. The high dimensions of image encoding compared to a relatively small number of training examples may lead to overfitting since the last FC regression layer must be capable of learning regression issues with a variety of independent degrees of problems including the localization error (translation and rotation) and generalization of unknown scenes, which has not been discussed in the original study of [18].

Recent work has shown the consideration of camera pose estimation without relying on video inputs but using single images [19]. The idea implemented using transferred features from deep learning model from both indoor [21] and outdoor datasets [22] and submitting these features to the support vector regression (SVR) for camera pose estimation provides less rotation and translation errors. However, for such a large dataset, it takes quite large amount of training time since the SVR is a regression method to preserve all the key features that define the functionality of the algorithm. In other words, the SVR attempts to ensure that the error in the estimation falls within a specified threshold. Hyperplane is utilized for predicting the results from multidimensional inputs. Searching for a hyperplane in a multidimensional space would increase the cost. Moreover, another difficulty is a variation of magnitudes, units, and range in real-world datasets. There are 7 camera pose parameters: 4 for rotation and 3 for translation are required; however, only one can be predicted at a time resulted in a large cost.

Building deep learning architecture from scratch requires a large amount of practice datasets and plenty of time to produce an efficient result. Transferring circumstances or things observed in one setting can help to improve the overall implications of another setting, resulting in less effort to prepare excessive amount of new practical datasets and processing time. Consequently, our research applies the transfer of single image features from in-depth learning of the VGG19 model to a proposed CNN in order to predict camera pose parameters in the form of rotation and translation. In the first phase, the local pattern in the input will be learnt through parametrizing each trained filter in the convolutional layers of the CNN. In other words, CNN is seeking to find the most appropriate way to predict the best result. In our study, for camera pose estimation, a performance evaluation of several architectures is conducted using 3 datasets where the first and second datasets are used to compare with [19], and the third one is used to compare with architectures in [18].
data.

2 PROPOSE ALGOLITHM

The next subsections provide instructions on how to insert figures, tables, and equations in your document.

2.1 Tables

This research presents camera pose estimation using CNN, which transfers knowledge from pretrained VGG19 model [23]. Deep transferred features from a single image are adopted to learn the relationship of each camera pose in terms of a 7-dimensional vector, where $\hat{t}$ is a 3D translation vector and $\hat{r}$ is a 4-dimensional rotation vector (quaternion).

Most of the traditional learning trained on small dataset and used in the specific task might not be as successful as expected especially for the unseen data since not enough retrained knowledge can be passed from one model to another. Transfer learning should enable us to utilize knowledge from previously learned tasks to newer, related ones. In the case of computer vision issues, other features of low quality such as edges, shapes, corners and intensity, can be transmitted across tasks, thereby enabling the transfer of knowledge across jobs. Therefore, as we have mentioned in the previous, information from an existing task serves as an additional input when learning a new goal.

To perform the transfer learning, many researchers have chosen VGG19 for their tasks. In [27, 28], Gatys et al. used the VGG19 through training approaches on object recognition for texture synthesis. Li et al. [29, 30] applied the VGG19 trained on ImageNet to extract hierarchical convolutional features for visual object tracking. Long et al. [31] have used VGG19 transfer learning to diagnose faults in the manufacturing industry. To achieve the characteristics of the desired image, in this research, we integrate transfer learning from the pre-trained VGG19 on the enormous ImageNet dataset used for 1000 category classification.

In this research, we demonstrate the feasibility of estimating the relationship between cameras via VGG19 transfer feature. The network begins with passing a 224 x 224 input image (RGB) through the stack of convolutional (conv.) and pooling layer to reduce the input's spatial dimensions (width x height) to a smaller size (downsampling). Nonlinear ReLU activation layers [38] are chosen to improve efficiency. Then, the three Full-Connected (FC) layers are trained to classify ImageNet Large Scale Visual Recognition Challenge (ILSVRC) into 1000 classes followed by the final softmax. Since our work does not require classification of data, the 4096 features dropped out from the fc7 layer of VGG19 are delivered to our camera pose estimation system. However, the resulting features have such a large dimension, which may cause data to be fragmented (sparse) and lead to the difficulty, time-consuming and inefficiency in the data analysis known as the curse of dimensionality [24]. The problem can be minimized by reducing the dimension of the data using the Latent Semantic Analysis (LSA) [25]. We obtain a 1000-dimensional feature vector as a result. Root mean square error (RMSE) [26] is chosen as our performance measurement matric to determine the rotation and translation prediction error. The overall workflow of our proposed camera pose estimation system is presented in Figure 1

In this research, we have developed a CNN model to create the appropriate features and to estimate the camera's relationship parameters for 3D reconstruction. The LSA reduced features are processed through our 4 convolutional layers and max pooling to simplify the network. The drop out layer is adopted to prevent overfit. Then, the dense layers are trained to support camera pose regression. The total of seven parameters are predicted for the rotation and translation of the cameras using two deep learning models, the rotation estimation model (four answers), and the translation estimation model (three answers).

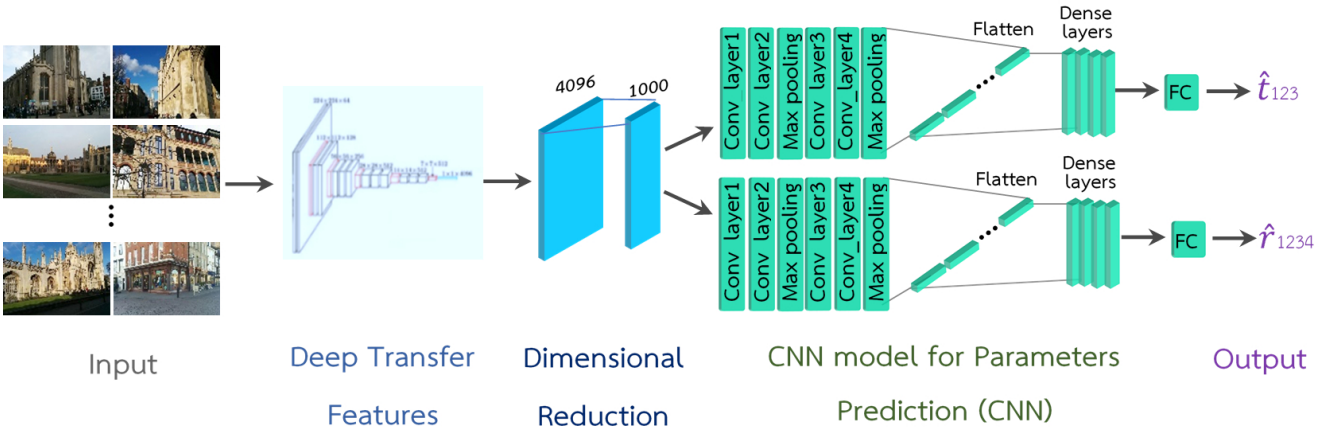

Figure 1: Overview of system operating process

Table 1: The average RMSE of our proposed algorithm compared with [19].

Train Data	Test Data	VGG+SVR [19]		Proposed (VGG+CNN)	
		Rotation (°)	Translation (m)	Rotation (°)	Translation (m)
Dome	Dome	0.45	2.45	0.07	1.15
Map		0.35	1.29	0.27	1.25
Map	Map	0.14	0.0035	0.03	0.0023
Dome		0.74	4.49	0.37	2.59
Average RMSE		0.44	2.25	0.19	1.25

3 EXPERIMENTS

In our experiments, the test dataset from 3 sources are used. The first one is indoor dome-shaped photography from CMU Panoptic Studio [21]. The image was taken by shooting around the object, with variety of rotation and translation. There are 209,934 images in dataset, divided into 146,957 training images and 62,977 test images. Each image comes with the extrinsic camera parameters; a set of 480 possible patterns of different individuals or groups movements such as moving, walking around and baseball swing etc. The second source is aerial photography [22] (Outdoor) taken from a camera attached to the drone which flew above 4 different locations - namely nadir square, stadthaus, rathaus, obelisk oblique square. Large translation will take place on X and Y axis, while Z axis is slightly different. The final source (Outdoor: Cambridge dataset) [32] consists of 6 sub-datasets, GreatCourt, KingsCollege, OldHospital, ShopFacade, StMarysChurch and Street. It is the popular dataset used to measure the efficiency of the algorithms.

The CNN model structure parameters and optimizer are evaluated in terms of the appropriate number of filter nodes {64, 128}, type of activation function [tanh, linear] and optimizer [adam and adadelta], batch size {32, 50, 100}, and the number of epochs {400, 500, 600} for achieving the optimum model structure with least RMSE value. The parameters that provide the lowest rotation (R) and translation (T) errors are considered to be the applied parameters for indoor (dome) and outdoor (Map and Cambridge) datasets. From experimental results, the best parameters obtained are 500 epochs, 32 batch size with adam optimizer and filter nodes of R {128,128,128,64}, T {64,64,64,128} for the indoor and and R {128,128,64,64}, T {128,128,128,64} for outdoor datasets, respectively. From the previously defined CNN structure, the experimental results of camera pose estimations of dataset 1 (Dome) and 2 (Map) are compared with the research results [19] as shown in Table 1

In order to assess the efficiency of the proposed model compared to [19], RMSEs are measured using the cases of dataset with the same view or shooting style as the training set while shooting different view or configuration as the test set. It is shown that the average RMSE of the rotation and translation of the proposed method training from dataset 1 (Indoor) and testing with the dataset 1 and 2 (outdoor), are 0.07, 0.27 degrees of rotation and 1.15, 1.25 meters of translation, respectively. Additionally, the average RMSE of rotation and translation is 0.03, 0.37 degrees and 0.0023, 2.59 meters respectively for the training with dataset 2 thus testing with the dataset of set 1 and set 2, which outperforms the methods[19] by 0.26 degrees and 1 meter, respectively. The experimental results show that the proposed model offers better average RMSE when training with dataset 1 than dataset 2 by 0.2 degrees of rotation and 0.1 meters of translation. This is due to the different amount of dataset 2, which is 100 times smaller than that of dataset 1 and the different in the shooting style of dataset 2 less variation than the dataset 1, resulting in insufficient cross-learning of the different

Table 2: Median translation (in meters) and rotation (in degrees) errors of different deep absolutes pose estimators, when tested on the Cambridge dataset.

Algorithm	GreatCourt	KingsCollege	OldHospital	ShopFacade	StMarysChurch	Street
PoseNet [16]	NA	1.97m, 5.40°	2.31m, 5.38°	1.46m, 8.08°	2.65m, 8.48°	3.67m, 6.50°
Dense PoseNet [16]	NA	1.66m, 4.86°	2.57m, 5.14°	1.41m, 7.18°	2.45m, 7.96°	2.96m, 6.00°
Bayesian PoseNet [32]	NA	1.74m, 4.06°	2.57m, 5.14°	1.25m, 7.54°	2.11m, 8.38°	2.14m, 4.96°
LSTM-Pose [20]	NA	0.99m, 3.65°	1.51m, 4.29°	1.18m, 7.44°	1.52m, 6.68°	NA
SVS-Pose [33]	NA	1.06m, 2.81°	1.50m, 4.03°	0.63m, 5.73°	2.11m, 8.11°	NA
PoseNet + Reprojection error pose loss [34]	7.00m, 3.7°	0.99m, 1.1°	2.17m, 2.9°	1.05m, 4.0°	1.49m, 3.40°	20.7m, 25.7°
VLocNet [35]	NA	0.836m, 1.42°	1.07m, 2.41°	0.593m, 3.53°	0.631m, 3.91°	NA
DSAC [36]	2.80m, 1.5°	0.30m, 0.5°	0.33m, 0.6°	0.09m, 0.40°	0.55m, 16°	NA
LearnLess(DSAC++) [37]	0.4m, 0.2°	0.18m, 0.3°	0.20m, 0.3°	0.06m, 0.30°	0.13m, 0.4°	NA
Active Search	NA	0.42m, 0.6°	0.44m, 1.0°	0.12m, 0.40°	0.19m, 0.5°	0.85m, 0.8°
Proposed	0.16m, 0.18°	0.20m, 0.21°	0.22m, 0.48°	0.11m, 0.20°	0.10m, 0.39°	0.77m, 0.76°

shooting. The third dataset is a comparison of the median rotation and translation errors with research [18]. Each study was trained and tested using its own dataset as shown in Table 2

Table 2 shows the implementation of the Cambridge dataset, which is popular for assessment the efficiency of the camera pose estimation algorithm. This dataset was taken with the angles of elevation that are the same plane as the location, which shooting style different from that of dataset 1 (indoor with the angles of press, parallel and elevation to the object) and dataset 2 (outdoor with the angles of press from drone flying parallel to the location). There are 6 subsets, subset 1 to 5 are locations and the last subset is the street. The last one is the most difficult combination of images since the whole image contains similar color and texture. In table 2, the results for the first 10 methods are based on [18], where LearnLess (DSAC++)[37] provides the least median error for subset 1 to 5. However, in [37], there is no result reported for the last subset. Nevertheless, it can be seen that the proposed method provides less median rotational and translation errors than those of the other methods. The sum of median error obtained from our proposed algorithm is less than [37] by 0.04 degrees rotation and 0.18 meters translation for subset 1 to 5 and less than the Active Search (SfM) by 0.04 degrees and 0.08 metres, respectively. From Tables 1 and 2, it is obvious that the our algorithm offers the best results in finding the rotation and translation in the shooting style similar to the dataset 2 with less average RMSE of 0.03 and 0.0023 degrees rotation compared with dataset 1 and 3, respectively.

4 CONCLUSION

This research presents the regression of camera pose estimation using the convolution neural network by transferring the basic features of VGG19 through LSA dimensional reduction technique prior to the input of the proposed CNN. In this research, the estimation of rotation and translation is derived from separated CNN models. According to the performance evaluation with a variety of dataset such as a different views, photography, brightness, blur and scrolling, the proposed model provided less overall predictive error than the other methods. The best performance is obtained from the dataset using the drone shooting down over the object. Nevertheless, even though the proposed model offers the least median error than the other methods for the Street subset, it provides the lowest performance compared to other subsets due to the ambiguity in color and texture of the scene.

REFERENCES

[1] Forster, C., Pizzoli, M., and Scaramuzza, D. 2014. SVO: Fast semi-direct monocular visual odometry. In: Intl. Conf. on Robotics and Automation (ICRA).
[2] Engel, J., Sturm, J., and Cremers, D. 2012. Camera-based navigation of a low-cost quadrocopter. In: Intl. Conf. on Intelligent Robot Systems (IROS).
[3] Achtelik, M., Weiss, S., and Siegwart, R. 2011. Onboard IMU and monocular vision based control for MAVs in unknown in- and outdoor environments. In: Intl. Conf. on Robotics and Automation (ICRA).
[4] Lim H., Sinha S. N., Cohen M. F., and Uyttendaele, M. 2012. Realtime image-based 6-dof localization in large-scale environments. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
[5] Lynen S., Sattler T., Bosse M., Hesch J., Pollefeys M., and Siegwart R.. 2015. Get Out of My Lab: Large-scale, Real-Time Visual-Inertial Localization. In Robotics: Science and Systems (RSS).
[6] Hartley R. I. and Zisserman A. 2004. Multiple View Geometry in Computer Vision. Cambridge University Press, ISBN: 0521540518, second edition.

[7] Bay H., Ess A., Tuytelaars T., and Van G. L. 2008. Speeded-up robust features (surf). Computer vision and image understanding, vol. 110, no. 3, pp. 346–359.
[8] Rublee E., Rabaud V., Konolige K., and Bradski, G. 2011, Orb: An efficient alternative to sift or surf. in Computer Vision, international conference on. IEEE, pp. 2564–2571.
[9] Mortensen, E. N., Deng, H., and Shapiro, L. 2005. A sift descriptor with global context. in Computer vision and pattern recognition, CVPR. IEEE computer society conference on, vol. 1. IEEE, pp. 184–190.
[10] Mahajan, D., Girshick, R., Ramanathan, V., He, K., Paluri, M., Li, Y., Bharambe, A. and van der Maaten, L., 2018. Exploring the limits of weakly supervised pretraining. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 181-196).
[11] Tan, M. and Le, Q.V., 2019. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. arXiv preprint arXiv:1905.11946.
[12] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A., 2015. Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1-9).
[13] Westlake, N., Cai, H. and Hall, P., 2016, October. Detecting people in artwork with CNNs. In European Conference on Computer Vision (pp. 825-841). Springer, Cham.
[14] Zhu, Y., Sapra, K., Reda, F.A., Shih, K.J., Newsam, S., Tao, A. and Catanzaro, B., 2019. Improving Semantic Segmentation via Video Propagation and Label Relaxation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 8856-8865).
[15] Badrinarayanan, V., Kendall, A. and Cipolla, R., 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence, 39(12), pp.2481-2495.
[16] Kendall, A., Grimes, M., and Cipolla, R. 2015. Posenet: A convolutional network for real-time 6-dof camera relocalization. In IEEE International Conference on Computer Vision (ICCV).
[17] Walch, F., Hazirbas, C., Leal-Taixe, L., Sattler, T., Hilsenbeck, S. and Cremers, D., 2017. Image-based localization using lstms for structured feature correlation. In Proceedings of the IEEE International Conference on Computer Vision (pp. 627-637).
[18] Shavit, Y., and Ferens, R. 2019. Introduction to Camera Pose Estimation with Deep Learning., arXiv preprint arXiv:1907.05272.
[19] Wattanacheep, Bh. ,and Chitsobhuk, O. 2019. Prediction of 3D rotation and translation from 2D images. In ICCCM 2019, July 27–29, 2019, Bangkok, Thailand. https://drive.google.com/file/d/1mE23Eg2x_4dHp7IKfQZNKhJoeXQXH1Be/view.
[20] Bell, S., Zitnick, C. L., Bala, K. ,and Girshick, R. 2016. Inside-Outside Net: Detecting Objects in Context with Skip Pooling and Recurrent Neural Networks. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR).
[21] Hanbyul J., Hyun S. P., Yaser Sh. 2014. MAP Visibility Estimation for Large-Scale Dynamic 3D Reconstruction. In Proceedings of CVPR, pp. 4321-4328.
[22] Nex, F., Gerke, M., Remondino, F., Przybilla H.-J., Bäumker, M., and Zurhorst, A., 2015. ISPRS Benchmark for Multi-Platform Photogrammetry. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. II-3/W4, pp.135-142.
[23] Karen, S. ,and Andrew, Z. 2015. Very Deep Convolutional Networks For large-scale Image Recognition. ICLR.
[24] Venkat, Naveen. 2018. The Curse of Dimensionality. Inside Out. 10.13140/RG.2.2.29631.36006.
[25] Landauer, T., *et al.* 2007. Handbook of Latent Semantic Analysis. Lawrence Erlbaum Associates.
[26] Alex, J. S, and Bernhard, S. 2004. A tutorial on support vector regression. Statistics and Computing 14, pp. 199–222.
[27] Dipanjan, S. 2018. A Comprehensive Hands-on Guide to Transfer Learning with Real-World Applications in Deep Learning.
[28] Gatys, Leon, Alexander, S. E., and Matthias, B. 2015. Texture synthesis using convolutional neural networks. In Advances in Neural Information Processing Systems, pp. 262-270.
[29] Bruna, Joan, Pablo, S., and Yann, L. 2015. Super-resolution with deep convolutional sufficient statistics. arXiv preprint arXiv:1511.05666.
[30] Li, Y., Yafei, Z., Yulong, X., Jiabao W., and Zhuang M. 2016. Robust scale adaptive kernel correlation filter tracker with hierarchical convolutional features. IEEE Signal Processing Letters 23, no. 8 : 1136-1140.
[31] Wen L., Li X., Li X., and Gao L. 2019. A New Transfer Learning Based on VGG-19 Network for Fault Diagnosis. 2019 IEEE 23rd International Conference on Computer Supported Cooperative Work in Design (CSCWD), Porto, Portugal, pp. 205-209.
[32] Kendall, A., and Cipolla, R. 2015. Modelling uncertainty in deep learning for camera relocalization. arXiv preprint arXiv:1509.05909.
[33] Naseer, T. ,and Burgard, W. 2017. Deep regression for monocular camera-based 6-dof global localization in outdoor environments. In 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 1525-1530). IEEE.
[34] Kendall, A. and Cipolla, R. 2017. Geometric loss functions for camera pose regression with deep learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 5974-5983).
[35] Valada, A., Radwan, N., and Burgard, W., 2018. Deep auxiliary learning for visual localization and odometry. In 2018 IEEE International Conference on Robotics and Automation (ICRA) (pp. 6939-6946). IEEE.
[36] Brachmann, E., Krull, A., Nowozin, S., Shotton, J., Michel, F., Gumhold, S., and Rother, C., 2017. DSAC-differentiable RANSAC for camera localization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 6684-6692). https://github.com/cvlab-dresden/DSAC.
[37] Brachmann, E., and Rother, C., 2018. Learning less is more-6d camera localization via 3d surface regression. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4654-4662). https://github.com/vislearn/LessMore
[38] Krizhevsky, A., Sutskever, I., and Hinton G. E. 2012. Imagenet classification with deep convolutional neural networks. In NIPS, pp. 1097–1105.