2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)
June 28-30, 2023. Seattle, Washington, USA

3-D Precision Positioning Based on Deep Comparison Convolutional Neural Networks

Bo-Xu Wen and Chih-Hung G. Li, *Member, IEEE*

***Abstract*— The UniShot 3D precision positioning model was developed using a deep comparison neural network (DCN). This dual-pipeline network extracts features from both the base and inquiry images in real time and predicts the observer's kinematic movements through internal comparison. We trained the model for transversal and depth movement detections and reported the precision and recall rates through static and dynamic experiments. We also analyzed the feature maps in the convolutional layers at various depths of the model to understand the comparison mechanism of the network. Results showed that the saliency feature patterns of DCNs are distinct from those of image recognition models and that the patterns for the transversal model were distinct from those for the depth model.**

I. Introduction

Object positioning holds great potential in industrial applications, where the precise spatial coordination of specific objects is crucial for processes like pick-and-place operations. Deep learning-based computer vision techniques have resulted in various object positioning methods, mainly using deep convolutional neural networks (ConvNets). Li and Chang's "OneShot" [1] visual positioning method showed promising results with a precision of ± 0.2 mm and a rotational error within $\pm 0.1°$. However, it requires scene-specific training, making it impractical for industrial use.

In this paper, we demonstrate the use of a deep comparison neural network (DCN) for precision positioning in transversal and depth movements, as shown in Fig. 1. The dual-pipeline ConvNet called UniShot addresses the limitations of prior methods by extracting critical spatial information through comparison of a base image and an inquiry image. This comparison network learns to identify key differences between the images, allowing it to predict precise coordinate values. As it can process any image pair, it is not limited to a specific scene and can be applied universally.

Our results show the accuracy and precision of DCN in both transversal and depth positioning. An analysis of the feature maps at various layers of the model was conducted to understand the comparison mechanisms in different applications. The results showed consistent image processing patterns that explain the comparison mechanisms of the network. This paper highlights the following significant contributions:

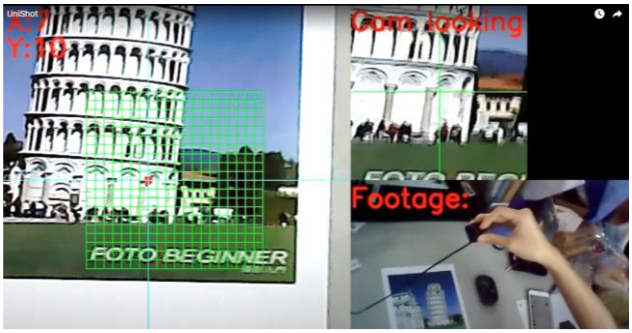

Fig. 1. The positioning DCN compares a base image with the real-time cam image to predict precise coordinates of the cam position.

1. Achieving high precision and accuracy in UniShot through training with 1.7 million image pairs.
2. An in-depth visual examination of the comparison mechanism in 3D precision positioning DCNs.
3. Experimental evidence based on static and dynamic tests.

II. Related Work

A. Object Localization

Visual localization tasks involve determining the relative position of a target object in an image or video based on the extracted information. Automated Optical Inspection (AOI) employs template matching to locate the target object's pixels [2]. Visual localization methods can be divided into local, global, and hybrid feature-based techniques [3]. Local methods like SIFT [4], SURF [5], and ORB [6] use neighboring pixels as descriptors to form a bag of words for semantic image description. These local methods have scale and orientation-invariant point features, allowing adaptability to changes in viewpoint and pose. Global methods represent the entire image as a high-dimensional signature, such as the Spatial Envelope [7], which defines perceptual dimensions for scene classification.

*This work was supported by the Ministry of Science and Technology of the Republic of China, Taiwan, under Contract No.: MOST 111-2221-E-027-106-MY2.

Bo-Xu Wen is with the Graduate Institute of Manufacturing Technology, National Taipei University of Technology, Taipei, 10608 Taiwan ROC (phone: +886-2-2771-2171 ext. 2080; fax: +886-2-2776-4889; e-mail: t110568023@ntut.org.tw).

Chih-Hung G. Li is with the Graduate Institute of Manufacturing Technology, National Taipei University of Technology, Taipei, 10608 Taiwan ROC (phone: +886-2-2771-2171 ext. 2092; fax: +886-2-2776-4889; e-mail: cL4e@mail.ntut.edu.tw).

Recent advancements in ConvNets have led to high-performance object localization methods, such as R-CNN [8-10] and YoLo [11-13]. R-CNN uses a two-step region proposal and classification approach, while YoLo employs a faster one-stage framework. ConvNets exhibit improved adaptability to viewpoint and illumination variations [14] and eliminate the need for manual parameter tuning in local-featured methods

OneShot [1, 15] is a ConvNet-based method for scene or object positioning that offers industrial precision, fast inference, high illumination tolerance, and ease of training. It trains a ConvNet classifier with a series of precisely framed images and predicts transversal and depth movements using sliding and size-variant frames. The method achieved pixel-level precision and was demonstrated in an industrial pick-and-place application [1]. A second version was developed with enhanced illumination tolerance through GAN-based template augmentation training [16]. However, it still requires specific training for each new application.

In the field of monocular depth estimation, state-of-the-art methods such as [17-19] utilize ConvNets and employ different learning techniques, including supervised, unsupervised, and semi-supervised learning. These methods aim to estimate an image's depth map with multiple objects at varying depths. In contrast, OneShot and UniShot focus on determining the uniform depth distance between the observer and the scene.

B. Deep Comparison Network

Comparison is a widely used measurement method, with template matching in AOI being a prime example, where the side-by-side comparison is performed by sliding a template over a base image for pixel-by-pixel comparison. However, this method is vulnerable to pixel offset and value changes, so comparison based on extracted features can be more reliable. DCN-based comparison networks were developed for object recognition, such as the Relation Network (RN) [20], which classifies objects by computing relation scores through a multi-input deep neural network that processes query images and image examples of each class. These networks use features learned at multiple levels of abstraction to enhance their performance in similarity problems [21].

DCN has also been used in various applications beyond object recognition, such as depth map generation from a pair of left and right RGB images [22, 23], virtual metrology for comparing surface appearance over time [24], and aesthetic ranking [25]. Despite their success, the understanding of the underlying mechanism of DCN is limited. In this paper, we use visual techniques such as feature maps to gain insights into the operating mechanism of our DCNs for 3-D precision positioning.

III. Positioning DCNs

We designed a dual-input DCN to estimate the transversal and depth movements of the observer through a side-by-side comparison of a pair of relevant images. The first image, referred to as the base image, contains the full view of the scene. In contrast, the second image, referred to as the inquiry image, contains a portion of the view resulting from the

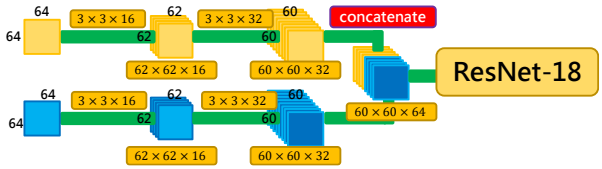

Fig. 2. Illustration of the dual-branch DCN architecture for transversal and depth positioning.

observer's transversal or depth movements. When the scene's depth is relatively uniform, such as a wall or a tabletop, the precise observer kinematics can be determined by comparing the contents of the two views before and after the movement.

To build the DCN, we collected and annotated pairs of base and inquiry images based on their revealed kinematic information. The training set W was formed with m samples, each consisting of an inquiry image and its corresponding base image,

$$W: \left(x_1, x^{B(x_1)}, y_1\right), \cdots, \left(x_m, x^{B(x_m)}, y_m\right) \in \mathbb{R}^q \times \mathbb{R}^q. \quad (1)$$

where the inquiry image x and the base image x^B have the same dimensionality q. The goal of the DCN is to classify each image pair into one of N categories,

$$\text{DCN}: \mathbb{R}^{2q} \longmapsto \{\sigma_1, \cdots, \sigma_N\}, \quad (2)$$

by predicting the category using the cross-entropy loss function,

$$\text{loss} = -\sum_i^N (y_i \times log(p_i)) . \quad (3)$$

where y_i denotes the truth label, and p_i denotes the predicted probability for the i^{th} class. The objective is to minimize the difference in probability between the network's output and the ground truth y_i, by adjusting the parameters θ of the DCN:

$$\theta^* = \underset{\theta}{argmin} \sum_i^m loss\left(\text{DCN}\left(x_i \oplus x^{B(x_i)}, \theta\right), y^i\right). \quad (4)$$

A. Network Architecture

The Dual-branch ConvNet architecture forms the basis of the DCN, as illustrated in Fig. 2. Both the base and inquiry images are resized to 64×64 pixels and fed into separate convolutional branches. Each branch comprises two convolutional layers for feature extraction, using 16 kernels in the first layer and 32 in the second. The output of both branches is then concatenated and inputted into a ResNet-18 pipeline for comparison computation. The number of output classes varied depending on the target item and desired resolution. For example, to predict the observer's transverse position across 21 different locations, a model with 21 output categories is needed.

B. Network Training

The performance of the positioning DCN depends on the training image set, which holds the kinematic relationships in each image pair. We use the automatic image processing procedure outlined in Fig. 3 to collect and annotate the images for our training set. The base images are resized to $2\Omega_X \times 2\Omega_Y$ pixels, then sub-images are systematically generated by

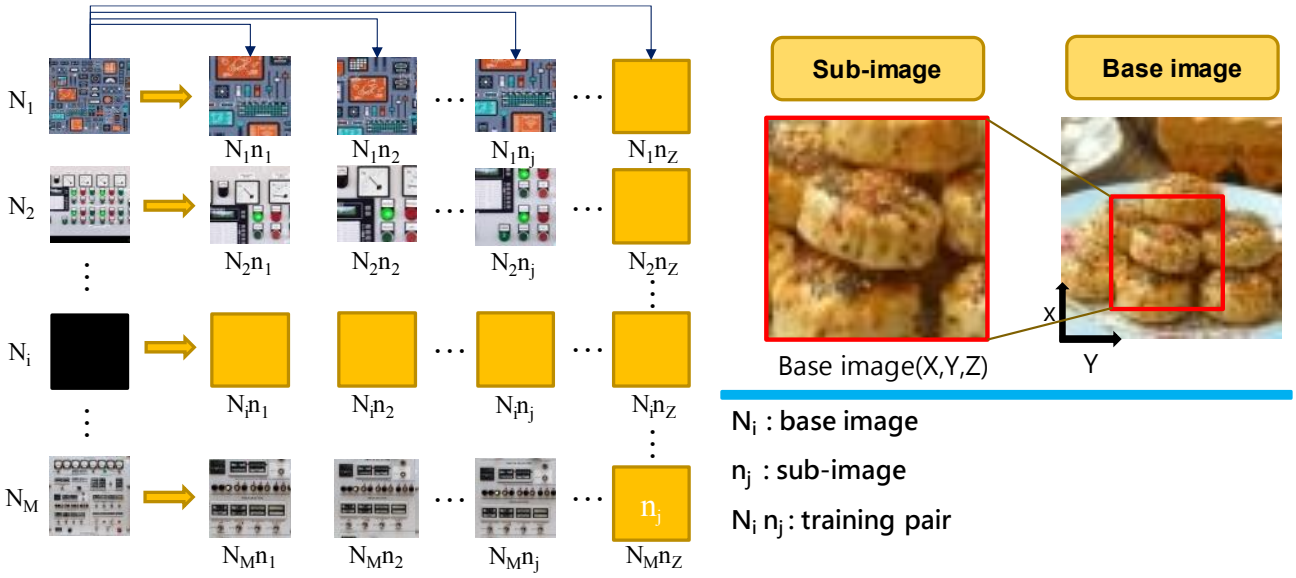

Fig. 3. Illustration of the training pair preparation process.

cropping the base images. The cropping frame moves horizontally to imply horizontal observer movement while reducing the frame size implies depth movement toward the observee. M base images were collected, generating M × Z training pairs. Increasing Z enhances detection accuracy, while increasing M increases generalization for better performance in unseen scenes.

The cropping frame size was defined as $w_X \times h_Y$ pixels and was moved horizontally in increments of $\mathcal{D}$ pixels and vertically in increments of $\mathcal{E}$ pixels. The frame size can be adjusted by incrementing $\mathcal{d}$ pixels, as shown in Fig. 4. The yellow dots symbolize the centers of the cropping frames, each with a range of sizes determined by $\mathcal{d}$. The area and position of the frame are determined by the frame center (X, Y), which is based on the horizontal movement $\beta\mathcal{D}$ and the vertical movement $\gamma\mathcal{E}$, and the amount of resizing $\alpha\mathcal{d}$. The total collection of the training images depends on the ranges of α, β, and γ as

$$W = \left(\sum_1^M x^B\right)\left[\sum_{\alpha=0}^{P}\sum_{\beta=0}^{Q}\sum_{\gamma=0}^{R} x(\alpha,\beta,\gamma)\right] \quad (5)$$

For a positioning model, one of the three variables is selected for detection, with the rest serving as supplementary variables for augmentation. In the case of depth detection, sub-images are annotated solely by α; however, the inclusion of β and γ enhances the tolerance to transverse variations. Our dynamic experiments confirmed that the integration of these variations leads to more precise predictions, particularly when dealing with unintentional camera movements.

To address potential degradation in performance due to illumination interference, we implemented data augmentation in the preparation of the training set. This was achieved by adding random patch noises to the sub-images through superimposing other images v with weighting ratios μ and ρ,

$$x \leftarrow \mu v + \rho x \quad (6)$$

where $0 \le \mu \le 0.7$, and $0.5 \le \rho \le 1.2$. Fig. 5 shows typical results. This operation simulates illumination noise and minor occlusions on the training images, improving the model's generalization ability. Our tests detailed below confirmed its effectiveness in real-world scenarios with varying illumination.

C. Performance Evaluation

We conducted two experiments to evaluate UniShot on a server computer with an Intel Core i7-7700 CPU, 64G RAM,

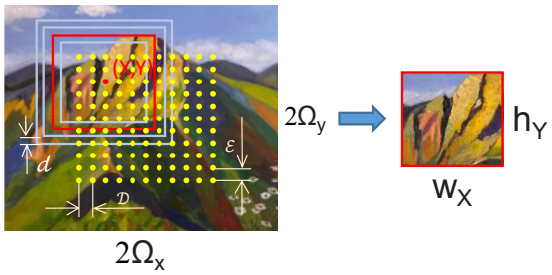

Fig. 4. Illustration of the cropping frame size and increments.

Fig. 5. The image blending operation for illumination enhancement.

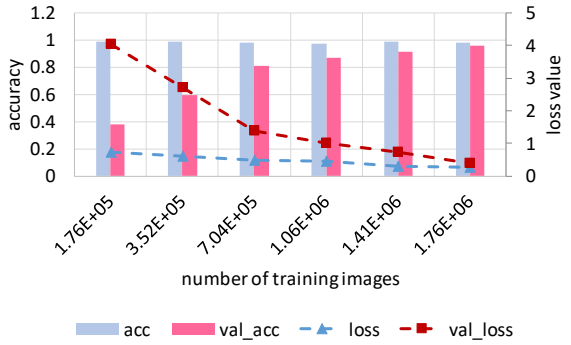

Fig. 6. Training and validation accuracy and loss values as functions of the training set size.

and an NVIDIA GeForce RTX 2080Ti GPU. The first experiment used static images, and the second used real-time images captured by a hand-held camera. The impact of training set size on transversal detection was investigated with a training set ranging from 176,000 to 1,760,000 images, as depicted in Fig. 6. As the number of training images increases, overfitting is significantly reduced, and the validation accuracy approaches the training accuracy, reaching close to 96% when 1,760,000 images were used.

In the static image experiment, we trained two transversal models using 1,760,000 training images and evaluated them on 4410 testing images. One model had 21 labels at a resolution of 10 pixels, while the other had 51 labels at a resolution of 4 pixels. The 21-label model had an average recall rate of 95.4% and an average precision rate of 95.2% (see Fig. 7). The 51-label model had a lower average recall rate of 79.1% and an average precision rate of 79.2%; however, the errors mainly were off by just one label. In the dynamic experiment, the 21-label model was tested, resulting in an average recall rate of 83.5% and an average precision rate of 83.5%. It is important to note that the testing images and scenes were different from those utilized for model training.

We trained two models with 1,760,000 training images in the depth detection experiment and evaluated them on 4410 testing images. One model had 13 labels and a resolution of 5

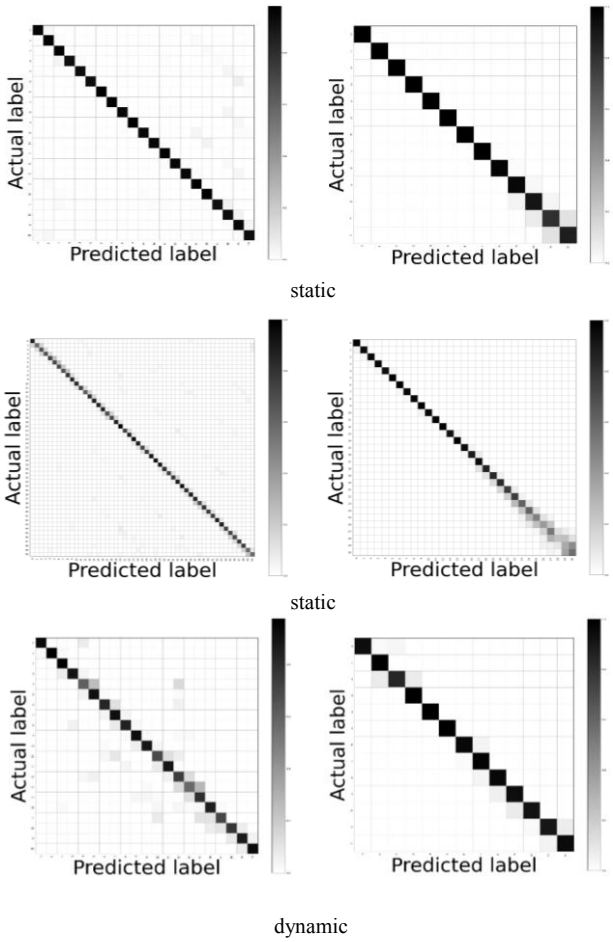

Fig. 7. Confusion matrices of the static and dynamic experiments of the positioning DCNs. *Left*: transversal detection; *Right*: depth detection.

pixels, while the other had 31 labels and a resolution of 2 pixels. The 13-label model had an average recall rate of 96.1% and an average precision rate of 96.2%. The 31-label model had a lower average recall rate of 81.3% and an average precision rate of 81.4%. In the dynamic experiment, the 13-label model had slightly lower recall at 95% and precision at

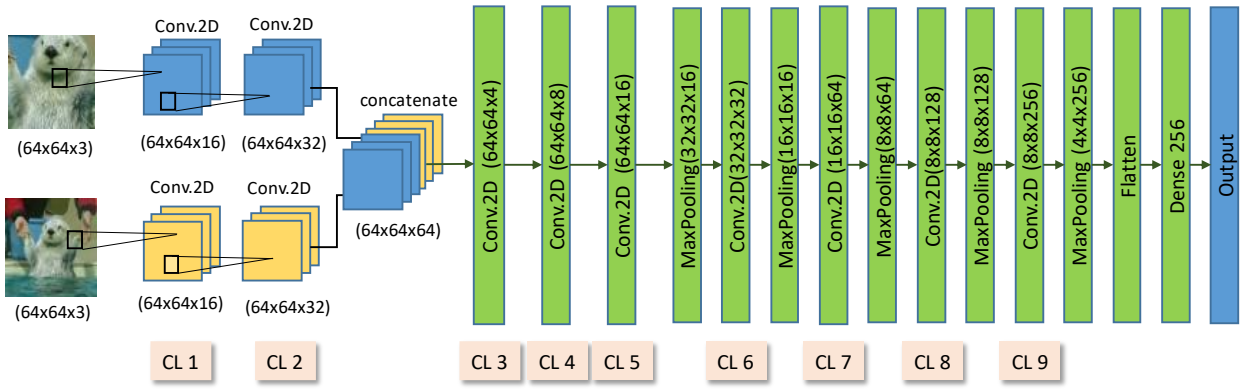

Fig. 8. Architecture of the feature map visualization model.

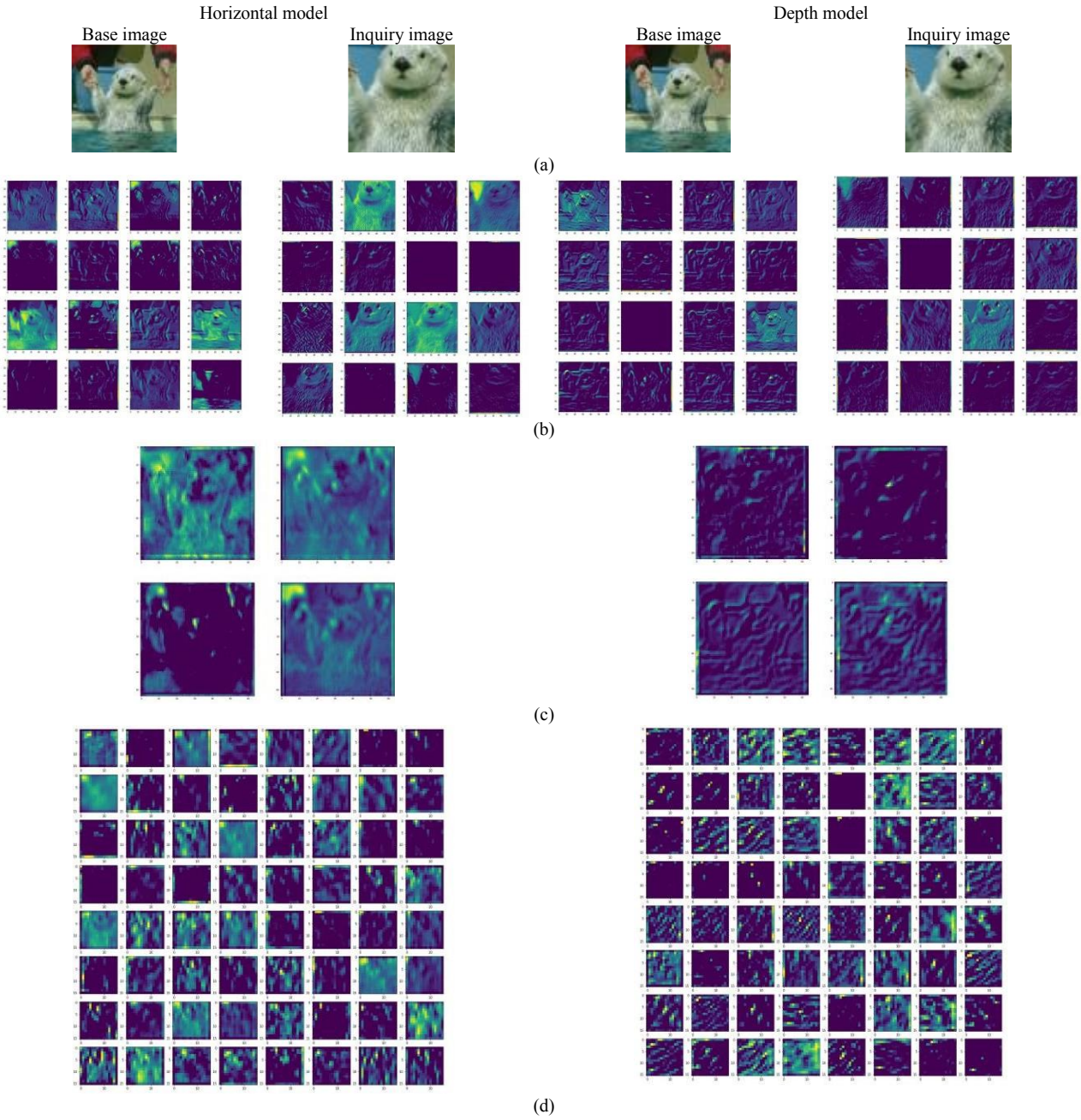

Fig. 9. Feature map comparisons between the horizontal and depth models: (a) input images, (b) first convolutional layers (CL1), (c) first convolutional layers after concatenation (CL3); (d) CL7.

94% compared to the static results, demonstrating its superior performance in handling dynamic variations.

IV. Visual Explanation

We constructed an 11-layer ConvNet using Keras and TensorFlow to visualize the activities in each convolutional layer, as illustrated in Fig. 8. The straightforward architecture facilitated the examination of the feature maps in each layer. However, the lack of skip connections, such as in ResNet, resulted in decreased performance for deeper models. The 12-layer model was the deepest that could yield optimal prediction results. We trained individual models for transverse and depth detection using the same training sets utilized for the earlier ResNet models. The testing accuracy for the 21-label transverse model was 65%, while the accuracy for the 13-label depth model was 91%, lower than the outcomes obtained from the ResNet models.

A. Early Features

Despite both performing spatial measurements, the internal processes of the transversal and depth models are expected to be distinct. This disparity is apparent from the earliest convolutional layers, as shown in Fig. 9 (b). The first layers in the horizontal model appear to detect color patches,

whereas those in the depth model focus on edge detection. This distinction continues after concatenating the outputs of both input pipelines, as depicted in the feature maps of Fig. 9 (c). The horizontal model highlights patches with different colors, while the depth model highlights high-contrast line structures.

B. Comparison Features

The dual pipelines in our models first extract essential features from the base and inquiry images; after concatenation, the single pipeline maps the features to the final output. Our observations showed that, at the output stage, the feature maps in the horizontal model evolved into vertical lines, while those in the depth model transformed into diagonal lines, as shown in Fig. 9 (d). This suggests that the horizontal model emphasizes horizontal differences, and the depth model prioritizes differences in object size over horizontal and vertical differences.

V. CONCLUSION

This study presents visual evidence of the mechanism of DCNs for precision positioning. Two DCNs were designed, one for transversal positioning and the other for depth positioning. The feature maps show that the network starts processing the information as early as the first convolutional layer during feature extraction from input images. The horizontal model focuses on color patches, while the depth model emphasizes edge detection. Further analysis of the feature maps after merging the base and query image pipelines shows that the horizontal model evolves into vertical lines, and the depth model transforms into diagonal lines. It is inferred that the vertical lines measure horizontal differences, while the diagonal lines measure size differences and provide depth information by disregarding both horizontal and vertical differences. The results exhibit distinct saliency patterns compared to those observed in traditional image recognition tasks.

REFERENCES

[1] C. G. Li and Y. -M. Chang, "Automated visual positioning and precision placement of a workpiece using deep learning," *Int. J. Adv. Manuf. Technol*, vol. 104, no. 9, pp. 4527–4538, 2019.

[2] F. Chen, et al., "Automated vision positioning system for dicing semiconductor chips using improved template matching method," *Int. J. Adv. Manuf. Technol*, vol. 100, pp. 2669–2678, 2019.

[3] N. Piasco, D. Sidibé, C. Demonceaux, and V. Gouet-Brunet, "A survey on visual-based localization: on the benefit of heterogeneous data," *Pattern Recognit.* vol. 74, pp. 90–109, 2018.

[4] S. Se, D. Lowe, and J. Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks," *Int. J. Robot. Research*, vol. 21, pp. 735–758, 2002.

[5] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: speeded up robust features," in *Comput. Vision – ECCV 2006, Lecture Notes Comp. Sci*. 3951, Springer, 2006.

[6] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *Proc. IEEE Int. Conf. Comput. Vision*, 2011, pp. 2564–2571.

[7] A. Oliva and A. Torralba, "Modeling the shape of the scene: a holistic representation of the spatial envelope," *Int. J. Comput. Vision*, vol. 42, no. 3, pp. 145–175, 2001.

[8] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2014, pp. 580–587.

[9] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vision (ICCV)*, 2015, pp. 1440–1448.

[10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell*, vol. 39, no. 6, pp. 1137–1149, 2016.

[11] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2016, pp. 779–788.

[12] J. Redmon, A. Farhadi, "YOLOv3: an incremental improvement," 2018, arXiv:1804.02767 [cs.CV].

[13] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: optimal speed and accuracy of object detection," 2020, arXiv:2004.10934 [cs.CV].

[14] N. Sunderhauf, S. Shirazi, F. Dayoub, B. Upcroft, and M. Milford, "On the performance of ConvNet features for place recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS)*, 2015, pp. 4297–4304.

[15] Y.-M. Chang, C. G. Li, and Y.-F. Hong, "Real-time object coordinate detection and manipulator control using rigidly trained convolutional neural networks," in *Proc. IEEE Int. Conf. Auto. Sci. Eng., (CASE)*, 2019, pp. 1347–1352.

[16] C. G. Li, Y.-H. Huang, "Deep-trained illumination-robust precision positioning for real-time manipulation of embedded objects," *Int. J. Adv. Manuf. Technol.*, vol. 111, pp. 2259-2276, 2020.

[17] A. Bhoi, "Monocular depth estimation: a survey," 2019, arXiv:1901.09402 [cs.CV].

[18] C. Godard, O. M. Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proc. Int. Conf. Comput. Vision (ICCV)*, 2019, pp. 3828-3838.

[19] I. Alhashim and P. Wonka, "High quality monocular depth estimation via transfer learning," 2019, arXiv:1812.11941 [cs.CV].

[20] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: relation network for few-shot learning," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2018, pp. 1199-1208.

[21] X. Zhang, Y. Qiang, F. Sung, Y. Yang and T. Hospedales, "RelationNet2: deep comparison network for few-shot learning," in *Proc. Int. Joint Conf. Neural Net. (IJCNN)*, 2020, pp. 1-8.

[22] J. Žbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches," *J. Mach. Learn. Res*, vol. 17, pp. 2287–2318, 2016.

[23] A. Kendall, et al., "End-to-end learning of geometry and context for deep stereo regression," in *Proc. IEEE Int. Conf. Comput. Vision (ICCV)*, 2017, pp. 66–75.

[24] C. -J. Yang, B. -X. Wen, and C. G. Li, "A deep comparison network for visual prognosis of a linear slide," in *Proc. Int. Conf. Adv. Robot. Intell. Syst. (ARIS)*, 2022, pp. 1-5.

[25] K. Ko, J. -T. Lee, and C. -S. Kim, "PAC-Net: pairwise aesthetic comparison network for image aesthetic assessment," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2018, pp. 2491-2495.

[26] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: visualising image classification models and saliency maps," 2013, arXiv:1312.6034 [cs.CV].

[27] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vision (ICCV)*, 2017, pp. 618-626.