Building Vision Transformers with Hierarchy Aware Feature Aggregation (Supplementary Materials)

Yongjie Chen[1,2] Hongmin Liu[1,2*] Haoran Yin[3] Bin Fan[1,2]
[1]School of Intelligence Science and Technology, University of Science and Technology Beijing
[2]Institute of Artificial Intelligence, University of Science and Technology Beijing
[3]Horizon Robotics
yongjie_chen@xs.ustb.edu.cn, hmliu_82@163.com, haoran.yin@horizon.ai, bin.fan@ieee.org

1. Detailed Architectures

The detailed model architecture is shown in Table 1, assuming an input image size of 224×224. Here, the Transformer Encoder used in PVT[5] is adopted. The meanings of the symbols are as follows:

- P_i : the patch size of Stage i.
- C_i : the channel number of the output of Stage i.
- L_i : the number of encoder layers in Stage i.
- R_i : the reduction ratio of the SRA in Stage i.
- N_i : the head number of the SRA in Stage i.
- E_i: the expansion ratio of the feed-forward layer in Stage i.
- $Cluster\ num$: The number of clusters in the SIA module.

Here, SRA refers to Spatial Reduction Attention proposed in PVT [5], which uses 2D convolution to downsample K and V separately during attention calculation to reduce the computational cost of attention.

2. Cluster Number

The number of clusters in the SIA module can be freely set, and different numbers of clusters will result in different degrees of feature aggregation, which may lead to different experimental results. We conducted experiments on the ImageNet-1k [1] dataset to show the influence of cluster number, and the results are shown in Table 2. From the results, it can be observed that when the number of clusters is 49, which is the conventional 1/4 down-sampling, the model's accuracy is the same as when the number of clusters is 81. However, when the number of clusters is 36, the model's accuracy drops by 0.5%. To have consistent feature map sizes as previous works, we set the number of clusters in the SIA module to 49.

*Hongmin Liu is the corresponding author

3. Early Stage Clustering

We designed a sliding window of size 2×2 to compare the clustering results with convolution with regards to the spatial extension. If all four patches within the window are assigned to the same cluster and adjacent windows have different classes, we consider the clustering result to be equivalent to convolution. After testing on the validation set of ImageNet-1k [1], we found that the similarity between using clustering and using convolution directly in the shallow layers (i.e., after stage 1 and stage 2) is as high as 95%, as shown in Figure 1.

4. Other Clustering Methods

To ensure a fair comparison with previous work [6], HAFA chose DPC-KNN [2] as the clustering algorithm. The advantage of DPC-KNN [2] is that it can automatically select cluster centers and requires fewer additional parameters. However, HAFA is compatible to other clustering algorithms. To show this, we replaced DPC-KNN [2] with the common K-means [4] algorithm and evaluated the performance on the ImageNet-1k dataset. The results are shown in Table 3. We chose the value of k to be 49 for a fair comparison and set the maximum number of iterations to be 10 for the consideration of speed.

5. Visualization

In this section, we will perform visualization for image classification, object detection, and semantic segmentation tasks. We used PVT-Tiny as our backbone and compared the results with and without the HAFA framework. First, we conducted image classification on the ImageNet-1k dataset, as shown in Figure 2. We visualized the clustering results and used CAM [7] to analyze the three patches with the most significant impact on the classification result. We can observe that after incorporating the HAFA framework, the model is able to capture the target object better. Secondly,

	Output Size	Layer Name	PVT-Tiny	PVT-Small	PVT-Large
Stage 1	56×56	Patch Embedding	$P_1 = 4; C_1 = 64$		
		Transformer Encoder	$\begin{bmatrix} R_1 = 8 \\ N_1 = 1 \\ E_1 = 8 \end{bmatrix} \times 2$	$\begin{bmatrix} R_1 = 8 \\ N_1 = 1 \\ E_1 = 8 \end{bmatrix} \times 3$	$\begin{bmatrix} R_1 = 8 \\ N_1 = 1 \\ E_1 = 8 \end{bmatrix} \times 3$
Stage 2	28×28	LAA	$P_2 = 2; C_2 = 128$		
		Transformer Encoder	$\begin{bmatrix} R_2 = 4 \\ N_2 = 2 \\ E_2 = 8 \end{bmatrix} \times 2$	$\begin{bmatrix} R_2 = 4 \\ N_2 = 2 \\ E_2 = 8 \end{bmatrix} \times 3$	$\begin{bmatrix} R_2 = 4 \\ N_2 = 2 \\ E_2 = 8 \end{bmatrix} \times 8$
Stage 3	14×14	LAA	$P_3 = 2; C_3 = 320$		
		Transformer Encoder	$\begin{bmatrix} R_3 = 2 \\ N_3 = 5 \\ E_3 = 4 \end{bmatrix} \times 2$	$\begin{bmatrix} R_3 = 2 \\ N_3 = 5 \\ E_3 = 4 \end{bmatrix} \times 6$	$\begin{bmatrix} R_3 = 2 \\ N_3 = 5 \\ E_3 = 4 \end{bmatrix} \times 27$
Stage 4	7×7	SIA	$Cluster\ num = 49; C_4 = 512$		
		Transformer Encoder	$\begin{bmatrix} R_4 = 1 \\ N_4 = 3 \\ E_4 = 4 \end{bmatrix} \times 2$	$\begin{bmatrix} R_4 = 1 \\ N_4 = 3 \\ E_4 = 4 \end{bmatrix} \times 3$	$\begin{bmatrix} R_4 = 1 \\ N_4 = 3 \\ E_4 = 4 \end{bmatrix} \times 3$

Table 1. **The detailed configuration of the model structure.** The LAA module in the HAFA framework is inserted between the first three stages, and the SIA module is inserted between stage 3 and stage 4.

Backbone	Cluster Number	Top-1 Acc(%)
PVT-Tiny[5]	36	77.0
	49	77.5
	81	77.5

Table 2. **Classification results for different numbers of clusters.**

Backbone	Clustering methods	Top-1 Acc(%)
PVT-Tiny[5]	DPC-KNN	77.5
	K-means	76.9

Table 3. **Results of different clustering methods.** K is chosen as 49, and the maximum number of iterations is set to 10.

we conducted object detection on the COCO2017 dataset [3], and the results are shown in Figure 3. Compared to the first column that uses the HAFA framework, the third column that does not use the HAFA framework exhibits obvious missed detections and false detections. Finally, we conducted semantic segmentation on the ADE20k dataset [8], and the results are shown in Figure 4. We can see that HAFA can cluster images based on the semantic information of objects, and some objects with similar semantic information are grouped together. This enables HAFA to better address some issues of missed or incorrect segmentations.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[2] Mingjing Du, Shifei Ding, and Hongjie Jia. Study on density peaks clustering based on k-nearest neighbors and principal component analysis. *Knowledge-Based Systems*, 99:135–145, 2016.

[3] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[4] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982.

[5] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 568–578, 2021.

[6] Wang Zeng, Sheng Jin, Wentao Liu, Chen Qian, Ping Luo, Wanli Ouyang, and Xiaogang Wang. Not all tokens are equal: Human-centric visual analysis via token clustering transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11101–11111, 2022.

[7] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016.

[8] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017.

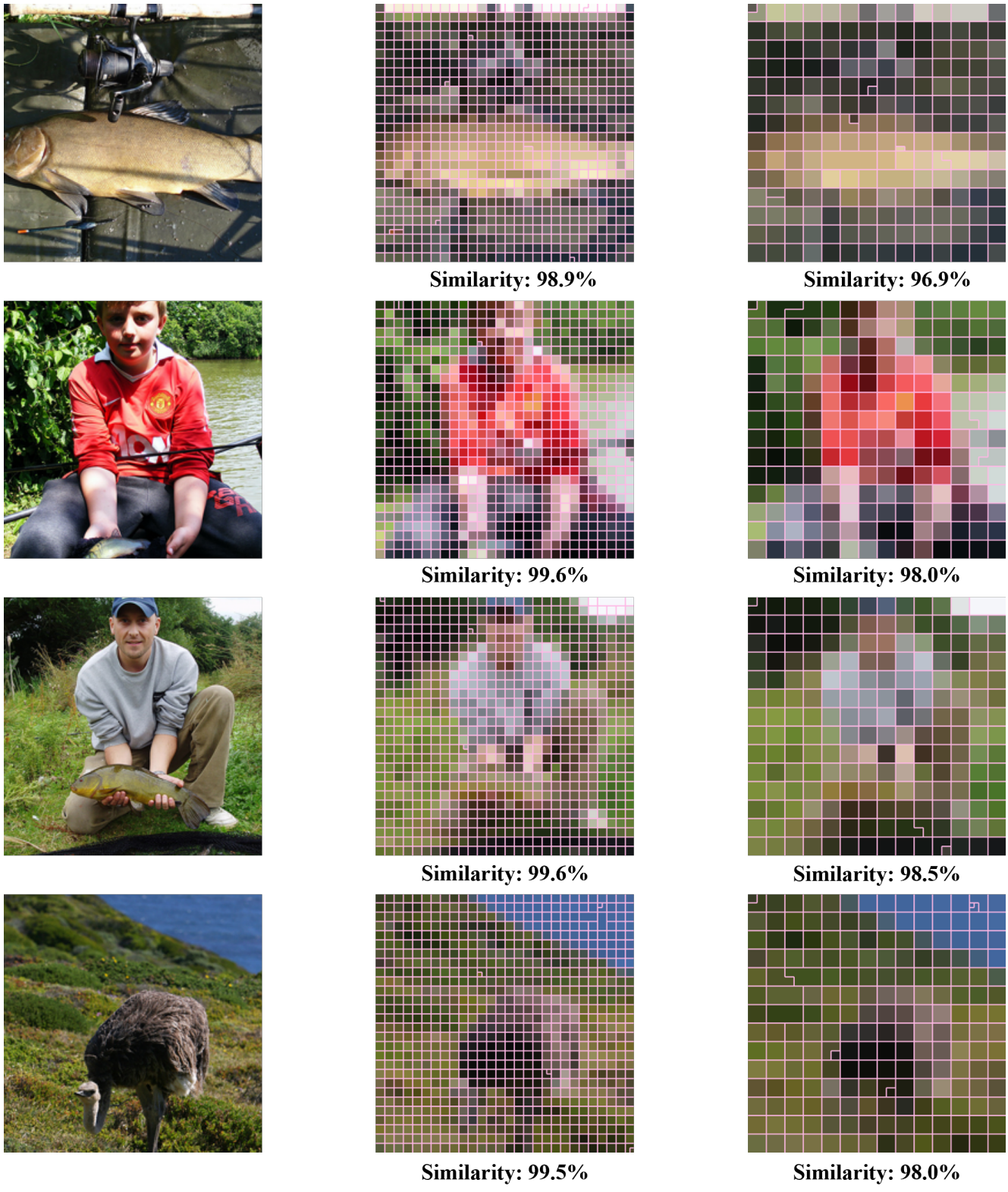

Figure 1. **Visualization of using clustering in the shallow layers.** Original image (the 1st column), The clustering results between Stage 1 and Stage 2 (the 2nd column), The clustering results between Stage 2 and Stage 3 (the 3rd column). The similarity between clustering results and convolution is marked at the bottom of the image.

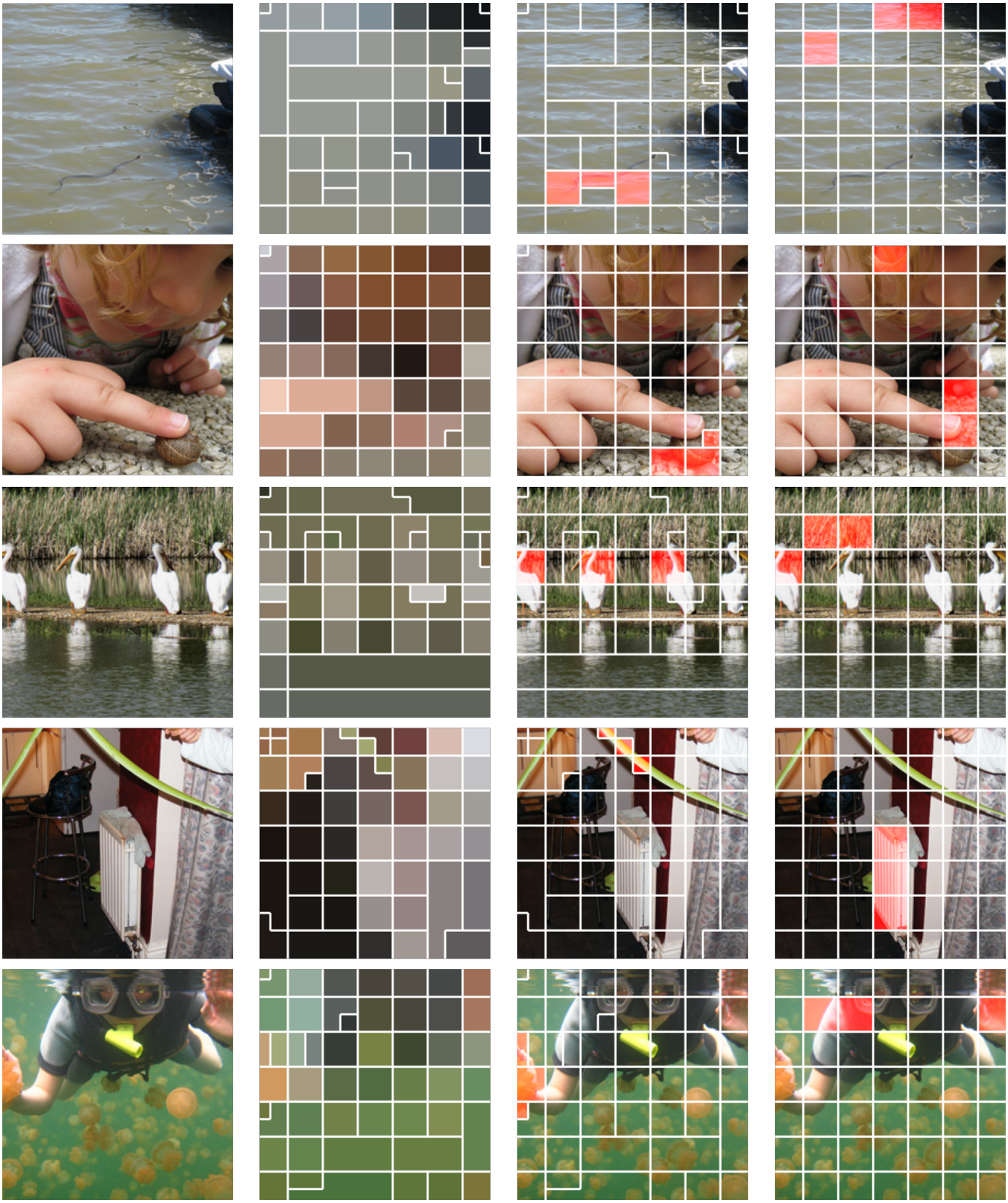

Figure 2. **Visualization of classification results.** Original image (the 1st column), Clustering results visualization (the 2nd column), The CAM visualization results with HAFA (the 3rd column) and without HAFA (the 4th column). The three patches with the most significant impact on the classification result are highlighted in red.

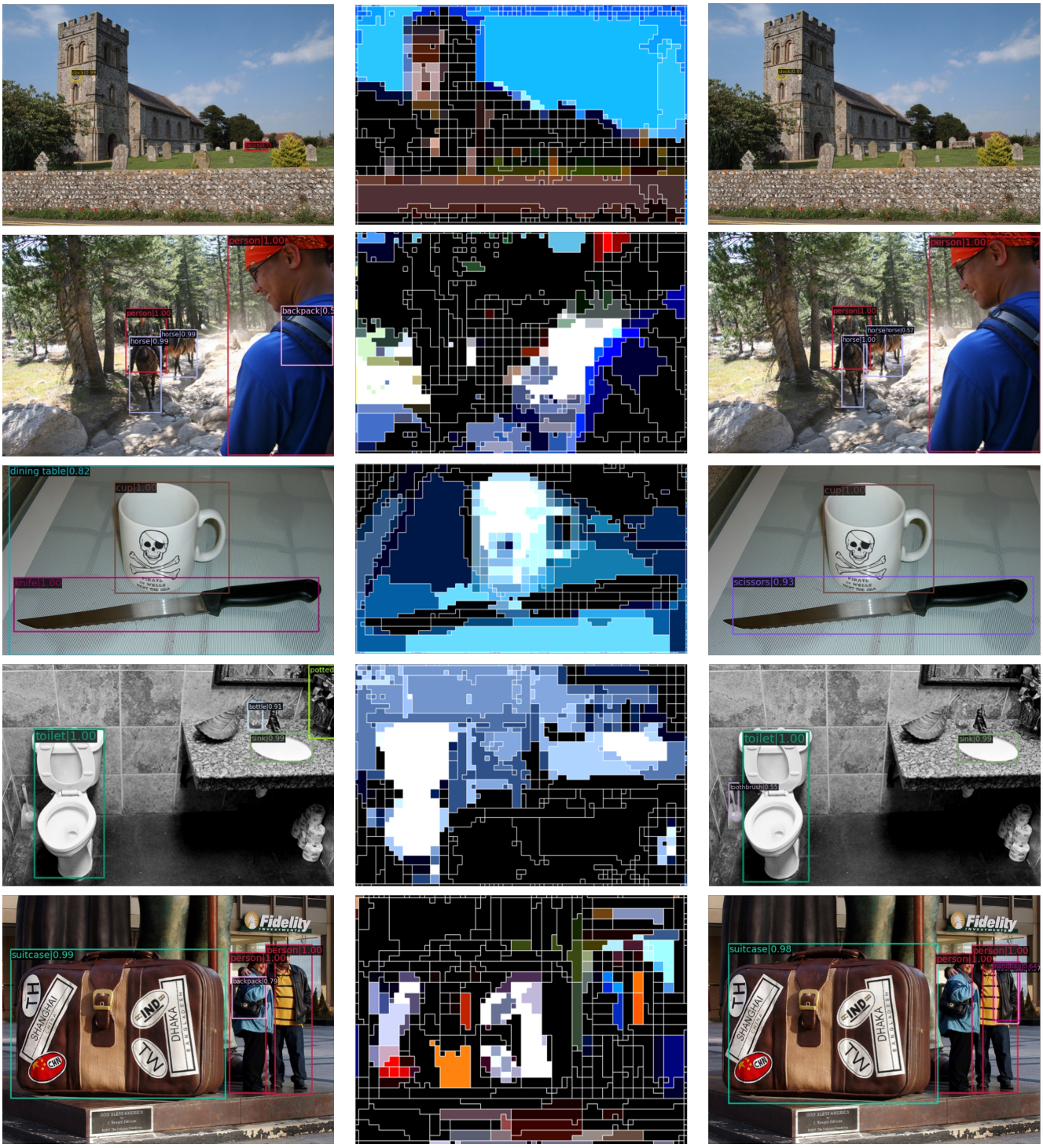

Figure 3. **Visualization of object detection.** The detection results using the HAFA framework (the 1st column), Clustering results visualization (the 2nd column), The detection results without using the HAFA framework (the 3rd column).

Figure 4. **Visualization of semantic segmentation.** Original image (the 1st column), Clustering results visualization (the 2nd column), The segmentation results with HAFA (the 3rd column) and without HAFA (the 4th column).