GazeCaps: Gaze Estimation with Self-Attention-Routed Capsules Supplementary Materials

Hengfei Wang[1*] Jun O Oh[2,3*] Hyung Jin Chang[1] Jin Hee Na[3] Minwoo Tae[2]
Zhongqun Zhang[1] Sang-Il Choi[2]

[1]University of Birmingham [2]Dankook University [3]VTouch, Inc

1. Detailed architecture of GazeCaps

Figure 1 illustrates the detailed architecture of our proposed GazeCaps. Given an input image, we first employ a ResNet-18 feature extractor and a convolutional layer to get image features. Then the features are reshaped to form primary capsules. Finally, we use two self-attention routing modules sequentially to get the gaze capsule which is also the predicted result.

2. Details of datasets

Table 1 summarises the statistics for each dataset.

3. Additional qualitative results

As shown in Figure 2 which shows additional qualitative results, our proposed method predicts more accurate gaze directions than the CNN-based method. What is notable is that we show the visualization of some challenging examples (dark environment and subject wearing glasses) in the last three rows. It shows that the CNN-based method usually focuses on wrong regions like forehead and mouth in these challenging cases while our proposed GazeCaps can find eye region accurately.

4. Implement details

We first pre-train our GazeCaps model on ETH-XGaze dataset. The training set in ETH-XGaze contains 765K images of 80 subjects. We use the last 4 subjects as the validation set, so 76 subjects are used as the pre-training set. For all training and testing, the size of the input image is 224x224. For pre-training, we use AdamW optimizer with learning rate = 1e-3, beta_1 = 0.9, and beta_2 = 0.999. In addition, we set a StepLR scheduler with 0.1 decay every 10 epochs. Pre-training takes 16 hours using 4 NVIDIA RTX A6000 GPU cards. For fine-tuning on EYEDIAP, Gaze360, MPIIFaceGaze, and RT-GENE, we use the same hyperparameters as the pre-training. It takes 8 hours (maximum) to converge on the same GPUs.

*Equal contribution

References

[1] Tobias Fischer, Hyung Jin Chang, and Yiannis Demiris. RT-GENE: Real-Time Eye Gaze Estimation in Natural Environments. In *European Conference on Computer Vision*, pages 339–357, September 2018. 2

[2] Kenneth Alberto Funes Mora, Florent Monay, and Jean-Marc Odobez. Eyediap database: Data description and gaze tracking evaluation benchmarks. Idiap-RR Idiap-RR-08-2014, Idiap, 5 2014. 2

[3] Petr Kellnhofer, Adria Recasens, Simon Stent, Wojciech Matusik, and Antonio Torralba. Gaze360: Physically unconstrained gaze estimation in the wild. In *IEEE International Conference on Computer Vision (ICCV)*, October 2019. 2

[4] Xucong Zhang, Yusuke Sugano, Mario Fritz, and Andreas Bulling. It's written all over your face: Full-face appearance-based gaze estimation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2299–2308, 2017. 2

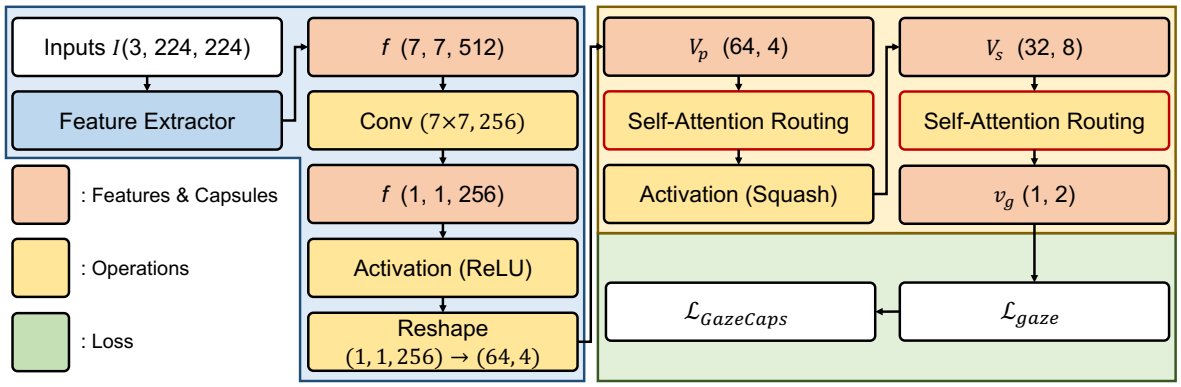

Figure 1. **Detailed network architecture of GazeCaps.** The framework of GazeCaps is divided into three parts: feature extraction part (blue zone), capsule part (yellow zone), and loss part (green zone). We present the shape of features and capsules in the framework.

	Subjects	Head Pose	Gaze	Data	Resolution
EYEDIAP [2]	16	±15°, 30°	±25°, 20°	237 min	HD & VGA
Gaze360 [3]	238	±90°, unknown	±140°, −50°	172,000	4096 × 3382
MPIIFaceGaze [4]	15	±15°, 30°	±20°, ±20°	45,000	1280 × 270
RT-GENE [1]	15	±40°, ±40°	±40°, −40°	122,531	1920 × 1080

Table 1. **Overview of the datasets used in experiments**. We show the number of subjects, the maximum head poses and gaze in horizontal and vertical directions in the camera coordinate systems, the amount of data (number of images or duration of the video), and image resolution

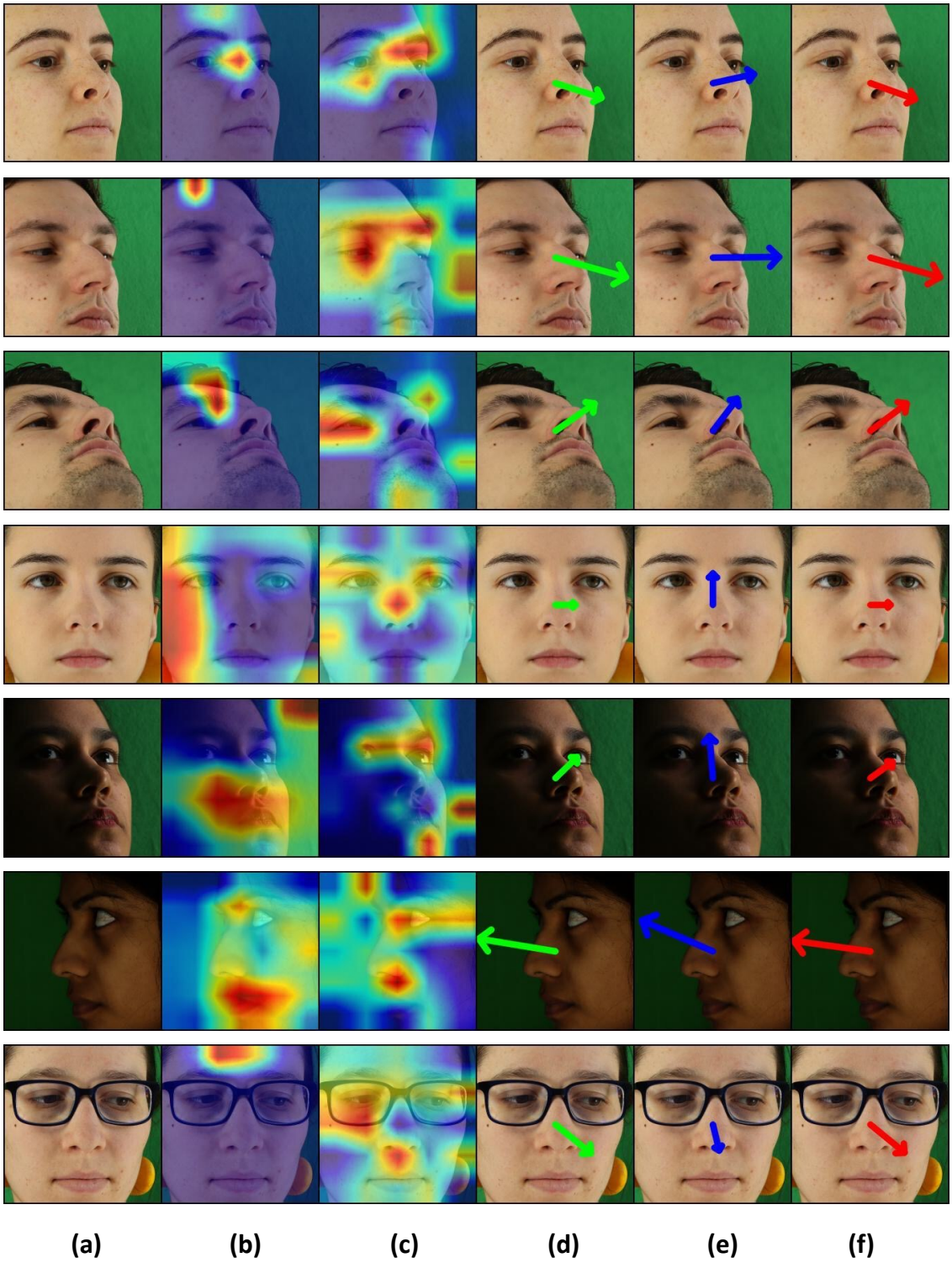

Figure 2. **Additional qualitative results**. (a) Input images, (b) Grad-CAM visualization of CNN-based method, (c) Grad-CAM visualization of GazeCaps, (d) Ground truth, (e) Prediction from CNN-based method, (f) Prediction from GazeCaps.