TASED-Net: Temporally-Aggregating Spatial Encoder-Decoder Network for Video Saliency Detection

Kyle Min Jason J. Corso
University of Michigan
Ann Arbor, MI 48109
{kylemin,jjcorso}@umich.edu

A. Clarification of the architecture

The table below summarizes the input-output sizes of our proposed TASED-Net architecture. As mentioned in the paper, the encoder network of TASED-Net consists of convolutional blocks from S3D [11], an inflated Inception-v1 (GoogLeNet) [8]. The details of 3D inception block are described in the S3D paper [11] and each name of the inception blocks (3a-5b) is from the original Inception paper [8]. Batch normalization [3] and ReLU [6] follow after each convolutional operations except the last layer. A sigmoid function is applied after the last convolution layer to produce an intensity map of saliency.

Type	Kernel / (Stride)	Input size	Output size	Description
Convolution	$7 \times 7 \times 7$ / (2, 2, 2)	$3 \times 32 \times 224 \times 384$	$64 \times 16 \times 112 \times 192$	First Conv-Block
Max-pool	$1 \times 3 \times 3$ / (1, 2, 2)	$64 \times 16 \times 112 \times 192$	$64 \times 16 \times 56 \times 96$	
Convolution	$1 \times 1 \times 1$ / (1, 1, 1)	$64 \times 16 \times 56 \times 96$	$64 \times 16 \times 56 \times 96$	
Convolution	$3 \times 3 \times 3$ / (1, 1, 1)	$64 \times 16 \times 56 \times 96$	$192 \times 16 \times 56 \times 96$	
Max-pool	$1 \times 3 \times 3$ / (1, 2, 2)	$192 \times 16 \times 56 \times 96$	$192 \times 16 \times 28 \times 48$	Max-pool in the main data stream
1st *Auxiliary pooling*	$4 \times 1 \times 1$ / (4, 1, 1)	$192 \times 16 \times 56 \times 96$	$192 \times 4 \times 56 \times 96$	Temporal-reduction
2nd *Auxiliary pooling*	$1 \times 3 \times 3$ / (1, 2, 2)	$192 \times 4 \times 56 \times 96$	$192 \times 4 \times 28 \times 48$	Switches-storing ($192 \times 4 \times 56 \times 96$)
Inception	Mixed / (1, 1, 1)	$192 \times 16 \times 28 \times 48$	$480 \times 16 \times 28 \times 48$	Mixed 3 Conv (two inception blocks: 3a, 3b)
Max-pool	$3 \times 3 \times 3$ / (2, 2, 2)	$480 \times 16 \times 28 \times 48$	$480 \times 8 \times 14 \times 24$	Max-pool in the main data stream
1st *Auxiliary pooling*	$4 \times 1 \times 1$ / (4, 1, 1)	$480 \times 16 \times 28 \times 48$	$480 \times 4 \times 28 \times 48$	Temporal-reduction
2nd *Auxiliary pooling*	$1 \times 3 \times 3$ / (1, 2, 2)	$480 \times 4 \times 28 \times 48$	$480 \times 4 \times 14 \times 24$	Switches-storing ($480 \times 4 \times 28 \times 48$)
Inception	Mixed / (1, 1, 1)	$480 \times 8 \times 14 \times 24$	$832 \times 8 \times 14 \times 24$	Mixed 4 Conv (five inception blocks: 4a-4e)
Max-pool	$2 \times 2 \times 2$ / (2, 2, 2)	$832 \times 8 \times 14 \times 24$	$832 \times 4 \times 7 \times 12$	Max-pool in the main data stream
1st *Auxiliary pooling*	$2 \times 1 \times 1$ / (2, 1, 1)	$832 \times 8 \times 14 \times 24$	$832 \times 4 \times 14 \times 24$	Temporal-reduction
2nd *Auxiliary pooling*	$1 \times 2 \times 2$ / (1, 2, 2)	$832 \times 4 \times 14 \times 24$	$832 \times 4 \times 7 \times 12$	Switches-storing ($832 \times 4 \times 14 \times 24$)
Inception	Mixed / (1, 1, 1)	$832 \times 4 \times 7 \times 12$	$1024 \times 4 \times 7 \times 12$	Mixed 5 Conv (two inception blocks: 5a, 5b)
Convolution	$1 \times 1 \times 1$ / (1, 1, 1)	$1024 \times 4 \times 7 \times 12$	$1024 \times 4 \times 7 \times 12$	Re-distribution of information
Transposed convolution	$1 \times 3 \times 3$ / (1, 1, 1)	$1024 \times 4 \times 7 \times 12$	$832 \times 4 \times 7 \times 12$	Spatial decoding
Max-Unpool	$1 \times 2 \times 2$ / (1, 2, 2)	$832 \times 4 \times 7 \times 12$	$832 \times 4 \times 14 \times 24$	Use the stored switches ($832 \times 4 \times 14 \times 24$)
Transposed convolution	$1 \times 3 \times 3$ / (1, 1, 1)	$832 \times 4 \times 14 \times 24$	$480 \times 4 \times 14 \times 24$	Spatial decoding
Max-Unpool	$1 \times 3 \times 3$ / (1, 2, 2)	$480 \times 4 \times 14 \times 24$	$480 \times 4 \times 28 \times 48$	Use the stored switches ($480 \times 4 \times 28 \times 48$)
Transposed convolution	$1 \times 3 \times 3$ / (1, 1, 1)	$480 \times 4 \times 28 \times 48$	$192 \times 4 \times 28 \times 48$	Spatial decoding
Max-Unpool	$1 \times 3 \times 3$ / (1, 2, 2)	$192 \times 4 \times 28 \times 48$	$192 \times 4 \times 56 \times 96$	Use the stored switches ($192 \times 4 \times 56 \times 96$) (Output size=Quarter-resolution)
Transposed convolution	$1 \times 4 \times 4$ / (1, 2, 2)	$192 \times 4 \times 56 \times 96$	$64 \times 4 \times 112 \times 192$	Spatial upsampling
Convolution	$2 \times 1 \times 1$ / (2, 1, 1)	$64 \times 4 \times 112 \times 192$	$64 \times 2 \times 112 \times 192$	Temporal-reduction
Transposed convolution	$1 \times 4 \times 4$ / (1, 2, 2)	$64 \times 2 \times 112 \times 192$	$4 \times 2 \times 224 \times 384$	Spatial upsampling
Convolution	$2 \times 1 \times 1$ / (2, 1, 1)	$4 \times 2 \times 224 \times 384$	$1 \times 1 \times 224 \times 384$	Temporal-reduction

Table 1: Detailed input-output sizes of our proposed TASED-Net architecture. First Conv-Block consists of three convolution layers and one max-pooling layer. Each name of the inception blocks (3a-5b) is from the original Inception paper [8]. Two sequential *Auxiliary poolings* first reduce the temporal dimension of the input feature map and then store the temporally-reduced switches for each unpooling layers.

B. Additional comparison

We apply our temporally-aggregating design to FCN [5], U-Net [7], and Deeplab [1] to further compare the performance and justify our architecture. All these architectures have achieved great success in dense prediction tasks. For the FCN and U-Net, we choose S3D [11] as the backbone. For the Deeplab, we first inflate ResNet-50 in 3D and pre-train it on Kinetics dataset [4], as presented by Wang *et al.* [10], and then apply two different versions of Deeplab [1, 2].

Method \ Metric	NSS	CC	SIM	AUC-J	s-AUC
FCN [5] (S3D)	2.435	0.440	0.329	0.890	0.702
U-Net [7] (S3D)	2.555	0.458	0.342	0.890	0.705
Deeplab-v2 [1] (ResNet-50 I3D)	2.382	0.430	0.339	0.893	0.689
Deeplab-v3+ [2] (ResNet-50 I3D)	2.406	0.434	0.335	0.892	0.700
TASED-Net (S3D)	**2.706**	**0.481**	**0.362**	**0.894**	**0.718**

Table 2: Comparison of other architectures (backbone network in bracket) with our temporally-aggregating scheme on the validation set of DHF1K [9]. It shows that TASED-Net with the proposed *Auxiliary pooling* is a more effective architecture for video saliency detection.

C. Denser TASED-Net

We provide an experimental comparison of TASED-Net and its denser (or deeper) version. We add two more transposed convolutional layers to each spatial decoding block in the prediction network of TASED-Net to build this version. The network size increases from 82MB to 118MB. It is shown that going deeper does not necessarily yield any performance gain, at least for our case.

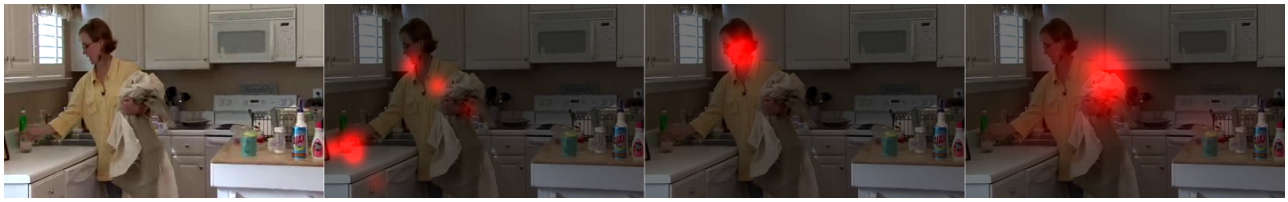
Method \ Metric	NSS	CC	SIM	AUC-J	s-AUC
TASED-Net	**2.706**	**0.481**	**0.362**	**0.894**	**0.718**
Denser TASED-Net	2.671	0.477	0.357	**0.894**	0.710

Table 3: Performance of TASED-Net and the denser version of it on the validation set of DHF1K [9] dataset. Denser TASED-Net yields slightly worse performance even if it has a larger number of parameters in the prediction network.

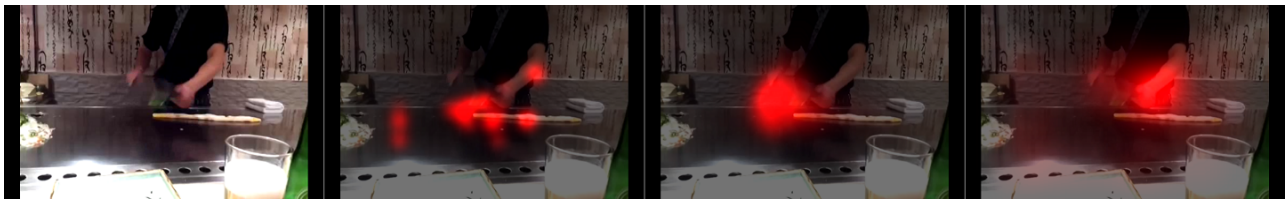
D. Additional qualitative results

We next provide additional qualitative results of TASED-Net and the previously leading state-of-the-art model ACLNet [9] on the validation set of the DHF1K dataset. All figures will play as videos when clicked in a suitable viewer (Adobe Reader, etc).

(a)



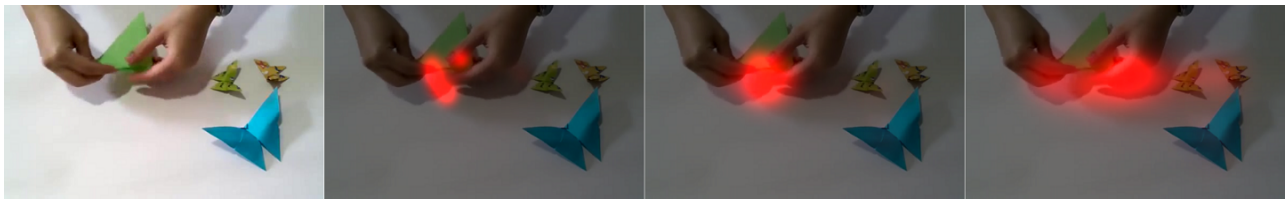
(b)



(c)

(d)

(e)



(f)

(g)

(h)

(i)

Figure 1: Qualitative results of our TASED-Net and the main competitive model ACLNet [9] on the DHF1K validation set (from left to right: Input video, Ground-truth, TASED-Net, and ACLNet). We show results on six videos for which our model outperforms ACLNet ((a)-(f)), and four videos for which ACLNet outperforms our model ((g)-(i)). As seen in (a)-(f), TASED-Net attends to the salient moving objects very well, even when there are many background objects. In (g)-(i), ground-truth fixation points are unstable and do not represent general human gaze behavior well; the number of fixation points are not enough to produce smooth ground-truth saliency maps for these videos, which suggests that saliency datasets should be created with more human subjects.

References

[1] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2018.

[2] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 801–818, 2018.

[3] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[4] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[5] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[6] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[8] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[9] Wenguan Wang, Jianbing Shen, Fang Guo, Ming-Ming Cheng, and Ali Borji. Revisiting video saliency: A large-scale benchmark and a new model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4894–4903, 2018.

[10] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7794–7803, 2018.

[11] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 305–321, 2018.