Unified Image and Video Saliency Modeling

Richard Droste[*], Jianbo Jiao[*], and J. Alison Noble

University of Oxford
{richard.droste, jianbo.jiao, alison.noble}@eng.ox.ac.uk

Abstract. Visual saliency modeling for images and videos is treated as two independent tasks in recent computer vision literature. While image saliency modeling is a well-studied problem and progress on benchmarks like SALICON and MIT300 is slowing, video saliency models have shown rapid gains on the recent DHF1K benchmark. Here, we take a step back and ask: Can image and video saliency modeling be approached via a unified model, with mutual benefit? We identify different sources of domain shift between image and video saliency data and between different video saliency datasets as a key challenge for effective joint modelling. To address this we propose four novel domain adaptation techniques— Domain-Adaptive Priors, Domain-Adaptive Fusion, Domain-Adaptive Smoothing and Bypass-RNN— in addition to an improved formulation of learned Gaussian priors. We integrate these techniques into a simple and lightweight encoder-RNN-decoder-style network, UNISAL, and train it jointly with image and video saliency data. We evaluate our method on the video saliency datasets DHF1K, Hollywood-2 and UCF-Sports, and the image saliency datasets SALICON and MIT300. With one set of parameters, UNISAL achieves state-of-the-art performance on all video saliency datasets and is on par with the state-of-the-art for image saliency datasets, despite faster runtime and a 5 to 20-fold smaller model size compared to all competing deep methods. We provide retrospective analyses and ablation studies which confirm the importance of the domain shift modeling. The code is available at https://github.com/rdroste/unisal.

Keywords: Visual saliency · Video saliency· Domain adaptation.

1 Introduction

When processing static scenes (images) and dynamic scenes (videos), humans direct their visual attention towards important information, which can be measured by recording eye fixations. The task of predicting the fixation distribution is referred to as *(visual) saliency prediction/modeling*, and the predicted distributions as *saliency maps*. Convolutional neural networks (CNNs) have emerged as the most performant technique for saliency modeling due to their capacity to learn complex feature hierarchies from large-scale datasets [2,20].

[*] Richard Droste and Jianbo Jiao contributed equally to this work.

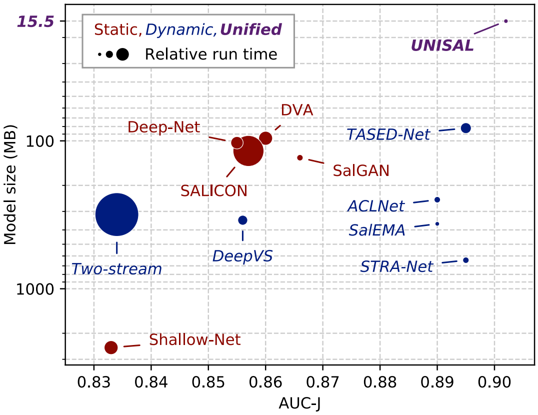

Fig. 1. Comparison of the proposed model with current state-of-the-art methods on the DHF1K benchmark [47]. The proposed model is more accurate (as measured by the official ranking metric AUC-J [5]) despite a model size reduction of 81% or more.

While most prior work focuses on image data, interest in video saliency modeling was recently accelerated through ACLNet, a dynamic saliency model that outperforms static models on the large-scale, diverse DHF1K benchmark [47]. However, as methods for video saliency modeling progress, it is usually considered a separate task to image saliency prediction [1,48,19,35,29,25] although both strive to model human visual attention. Current dynamic models use image data only for pre-training [1,19,35,29,25] or auxiliary loss functions [47]. In addition, many dynamic models are incompatible with image inputs since they require optical flow [1,25] or fixed-length video clips for spatio-temporal convolutions [19,35]. In this paper, we ask the question: *Is it possible to model static and dynamic saliency via one unified framework, with mutual benefit?*

First, we present experiments that identify the domain shift between image and video saliency data and between different video saliency datasets as a crucial hurdle for joint modelling. Consequently, we propose suitable domain adaptation techniques for the identified sources of domain shift. To study the benefit of the proposed techniques, we introduce the UNISAL neural network, which is designed to model visual saliency on image and video data coequally while aiming for simplicity and low computational complexity. The network is simultaneously trained on three video datasets—DHF1K [47], Hollywood-2 and UCF-Sports [34]—and one image saliency dataset, SALICON [20].

We evaluate our method on the four training datasets, among which DHF1K and SALICON have held-out test sets. In addition, we evaluate on the established MIT300 image saliency benchmark [21]. We find that our model significantly outperforms current state-of-the-art methods on all video saliency datasets and achieves competitive performance for the image saliency datasets, with a fraction of the model size and faster runtime than competing models. The performance of UNISAL on the challenging DHF1K benchmark is shown in Figure 1. In summary, our contributions are as follows:

- To the best of our knowledge, we make the first attempt to model image and video visual saliency with one unified framework.
- We identify different sources of domain shift as the main challenge for joint image and video saliency modeling and propose four novel domain adaptation techniques to enable strong shared features: Domain-Adaptive Priors, Domain-Adaptive Fusion, Domain-Adaptive Smoothing, and Bypass-RNN.
- Our method achieves state-of-the-art performance on all video saliency datasets and is on par with the state-of-the-art for all image saliency datasets. At the same time, the model achieves a 5 to 20-fold reduction in model size and faster runtime compared to all existing deep saliency models.

2 Related Work

Image Saliency Modeling. Most visual saliency modeling literature aims to predict human visual attention mechanisms on static scenes. Early saliency models [17,3,42,13,26,22] focus on low-level image features such as intensity/contrast, color, edges, *etc.*, and are are therefore referred to as *bottom-up* methods. Recently, the field has achieved significant performance gains through deep neural networks and their capacity to learn high-level, *top-down* features, starting with Vig *et al.* [45] who propose the first neural network-based approach. Jiang *et al.* [20] collect a large-scale saliency dataset, SALICON, to facilitate the exploration of deep learning-based saliency modeling. Zheng *et al.* [51] investigate the impact of high-level observer tasks on saliency modeling. Other papers mainly focus on network architecture design with increasing model sizes. For instance, Pan *et al.* [37] evaluate shallow and deep CNNs for saliency prediction, and Kruthiventi *et al.* [23] introduce dilated convolutions and Gaussian priors into the VGG network architecture. Kuemmerer *et al.* [24] propose a simplified VGG-based network while Wang *et al.* [46] add skip connections to fuse multiple scales and Cornia *et al.* [7] add an attentive convolutional LSTM and learned Gaussian priors. Yang *et al.* [50] expand on the idea of dilated convolutions based on the inception network architecture. While exploration is still ongoing for image saliency modeling, dynamic scenes are arguably at least as relevant to human visual experience, but have received less attention in the literature to date.

Video Saliency Modeling. Similar to image saliency models, early dynamic models [33,32,39,15] predict video saliency based on low-level visual statistics, with additional temporal features (*e.g.*, optical flow). Marat *et al.* [33] use video frame pairs to compute a static and a dynamic saliency map, which are fused for the final prediction. Marat *et al.* [33] and Zhong *et al.* [52] combine spatial and temporal saliency features and fuse the predictions. By extending the center-surround saliency in static scenes, Mahadevan *et al.* [32] use dynamic textures to model video saliency. The performance of these early models is limited by the ability of the low-level features to represent temporal information.

Consequently, deep learning based methods have been introduced for dynamic saliency modeling in recent years. Gorji *et al.* [10] propose to incorporate attentional push for video saliency prediction, via a multi-stream convolutional long short-term memory network (ConvLSTM). Jiang *et al.* [19] show that human attention is attracted to moving objects and propose a saliency-structured ConvLSTM to generate video saliency. A recent work [48] presents a new large-scale video saliency dataset, DHF1K, and propose an attention mechanism with ConvLSTM to achieve better performance than static deep models. The DHF1K dataset, sparked advances [35,25,29] in video saliency prediction, exploring different strategies to extract temporal features (optical flow, 3D convolutions, different recurrences). However, the above methods either extend prior image saliency models or focus on video data alone with limited applicability to static scenes. Guo *et al.* [11] present a spatio-temporal model that predicts image and video saliency through the phase spectrum of the Quaternion Fourier Transform but the model lacks the necessary high-level information for accurate saliency prediction. While a recent learning-based approach [30] extends the image domain to the spatio-temporal domain by using LSTMs, such models are specialized for video data, rendering them unable to simultaneously model image saliency.

Domain Adaptation. We focus on domain specific learning, a form of domain adaptation which enables a learning system to process data from different domains by separating domain-invariant (shared) and domain-specific (private) parameters [6]. Domain Separation Networks (DSN) [4], for instance, are autoencoders with additional private encoders. Instead of an autoencoder, Tsai *et al.* [43] introduce an adversarial loss that enforces shared and private encoders networks. Xiao *et al.* [49] propose Domain Guided Dropout that results in different sub-networks for each domain, and Rozantev *et al.* [38] train entirely separate networks for each domain, coupled through a similarity loss. In contrast to using separate networks, the AdaBN method [28] adjusts the batch-normalization (BN) parameters of a shared network based on samples from a given target domain. The DSBN method [6] generalizes this idea by training a separate set of BN parameters for each domain. In general, these existing methods result in a large proportion of domain-specific parameters. In contrast, we propose domain-adaptation techniques that are aimed to bridge the domain gap of saliency datasets with a maximum proportion of shared parameters.

3 Unified Image and Video Saliency Modeling

3.1 Domain-Shift Modeling

In this section we present analyses to examine the domain shift between image and video data and between different video saliency datasets. We use the insights to design corresponding domain adaptation methods. Following Wang *et al.* [48], we select the video saliency datasets DHF1K [48], Hollywood-2 and UCF Sports [34], and the image saliency dataset SALICON [20].

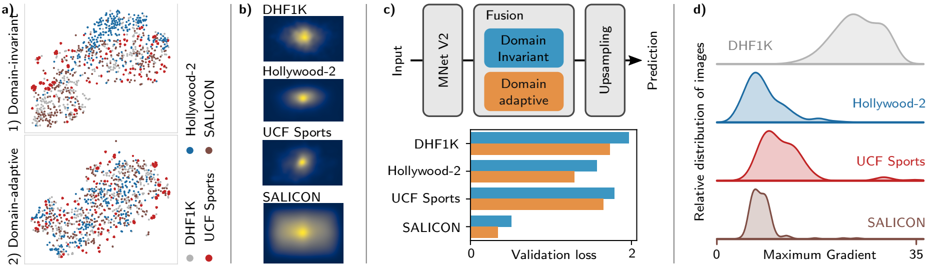

Fig. 2. Experiments to examine the domain shift between the saliency datasets. **a)** t-SNE visualization of MNet V2 features after domain-invariant and domain-adaptive normalization. **b)** Average ground truth saliency maps. **c)** Comparison of validation losses when training a simple saliency model with domain-invariant and domain-adaptive fusion. **d)** Distributions of ground truth saliency map sharpness.

Domain-Adaptive Batch Normalization. Batch normalization (BN) aims to reduce the internal covariate shift of neural network activations by transforming their distribution to zero mean and unit variance for each training batch. Simultaneously, it computes running estimates of the distribution mean and variance for inference. However, estimating these statistics across different domains results in inaccurate intra-domain statistics, and therefore a performance trade-off. In order to examine the domain shift between the datasets, we conduct a simple experiment: We randomly sample 256 images/frames from each dataset and compute their average pooled MobileNet V2 (MNet V2) features. We then visualize the distribution of the feature vectors via t-SNE [31] after normalizing them with the mean and variance of 1) all samples (domain-invariant) or 2) the samples from the respective dataset (domain-adaptive). The results, shown in Figure 2 a), reveal a significant domain shift among the different datasets, which is mitigated by the domain-adaptive normalization. Consequently, we employ *Domain-Adaptive Batch Normalization* (DABN), *i.e.*, a different set of BN modules for each dataset. During training and inference, each batch is constructed with data from one dataset and passed through the corresponding BN modules.

Domain-Adaptive Priors. Figure 2 b) shows the average ground truth saliency map for each training dataset. Among the video datasets, Hollywood-2 and UCF Sports exhibit the strongest center bias, which is plausible since they are biased towards certain content (movies and sports) while DHF1K is more diverse. SALICON has a much weaker center bias than the video saliency datasets, which can potentially be explained by the longer viewing time of each image/frame (5 s *vs*. 30 ms to 42 ms) that allows secondary stimuli to be fixated. Accordingly, we propose to learn a separate set of Gaussian prior maps for each dataset.

Domain-Adaptive Fusion. We hypothesize that similar image features can have varying visual saliency for images/frames from different training datasets.

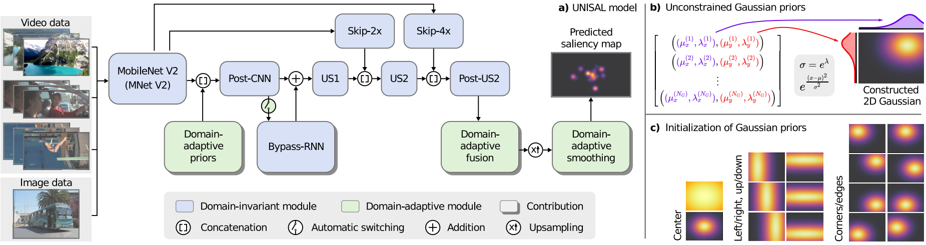
Video data
Image data
MobileNet V2 (MNet V2)
Skip-2x
Skip-4x
Post-CNN
US1
US2
Post-US2
Domain-adaptive priors
Bypass-RNN
Domain-adaptive fusion
Domain-adaptive smoothing
a) UNISAL model
Predicted saliency map
Domain-invariant module
Domain-adaptive module
Contribution
Concatenation
Automatic switching
Addition
Upsampling
b) Unconstrained Gaussian priors
$\left[\begin{array}{c}\left((\mu_x^{(1)},\lambda_x^{(1)}),(\mu_y^{(1)},\lambda_y^{(1)})\right)\\ \left((\mu_x^{(2)},\lambda_x^{(2)}),(\mu_y^{(2)},\lambda_y^{(2)})\right)\\ \vdots \\ \left((\mu_x^{(N_G)},\lambda_x^{(N_G)}),(\mu_y^{(N_G)},\lambda_y^{(N_G)})\right)\end{array}\right]$
$\sigma = e^{\lambda}$
$e^{\frac{(x-\mu)^2}{\sigma^2}}$
Constructed 2D Gaussian
c) Initialization of Gaussian priors
Center
Left/right, up/down
Corners/edges

Fig. 3. **a)** Overview of the proposed framework. The model consists of a MobileNet V2 (MNet V2) encoder, followed by concatenation with learned Gaussian prior maps, a *Bypass-RNN*, a decoder network with skip connections, and *Fusion* and *Smoothing* layers. The prior maps, fusion, smoothing and batch-normalization modules are domain-adaptive in order to account for domain-shift between the image and video saliency datasets and enable high-quality shared features. **b)** Construction of the prior maps from learned Gaussian parameters. **c)** Prior maps initialization.

For example, the Hollywood-2 and UCF Sports datasets are *task-driven*, *i.e.*, the viewer is instructed to identify the main action shown. On the other hand, the DHF1K and SALICON datasets contains *free-viewing* fixations. To test the hypothesis, we design a simple saliency predictor (see Figure 2 c): The outputs of the MNet V2 model are fused to a single map by a *Fusion* layer (1×1 convolution) and upsampled through bilinear interpolation. We train the *Fusion* layer until convergence with 1) one set of weights (domain-invariant) or 2) different weights for each dataset (domain-adaptive). We find that the validation loss is lower for all datasets for setting 2), where the network can weigh the importance of the feature maps differently for each dataset. Consequently, we propose to learn a different set of *Fusion* layer weights for each dataset.

Domain-Adaptive Smoothing. The size of the blurring filter which is used to generate the ground truth saliency maps from fixation maps can vary between datasets, especially since the images/frames are resized by different amounts. To examine this effect, we compute the distribution of the ground truth saliency map sharpness for each dataset. Sharpness is computed as the maximum image gradient magnitude after resizing to the model input resolution. The results in Figure 2 d) confirm the heterogeneous distributions across datasets, revealing the highest sharpness for DHF1K. Therefore, we propose to blur the network output with a different learned *Smoothing* kernel for each dataset.

3.2 UNISAL Network Architecture

We introduce a simple and lightweight neural network architecture termed *UNISAL* that is designed to model image and video saliency coequally and implements the proposed domain-adaptation techniques. The architecture, illustrated in Figure 3, follows an encoder-RNN-decoder design tailored for saliency modeling.

Encoder Network. We use MobileNet-V2 (MNet V2) [40] as our backbone encoder for three reasons: First, its small memory footprint enables training with sufficiently large sequence length and batch size; second, its small number of floating point operations allows for real time inference; and third, we expect the relatively small number of parameters to mitigate overfitting on smaller datasets like UCF Sports. The main building blocks of MNet V2 are *inverted residuals*, *i.e.*, sequences of pointwise convolutions that decompress and compress the feature space, interleaved with depthwise separable 3×3 convolutions. Overall, for an input resolution of $[r_x, r_y]$, MNet V2 computes feature maps at resolutions of $\frac{1}{2^\alpha}[r_x, r_y]$ with $\alpha \in \{1, 2, 3, 4, 5\}$. The output has 1280 channels and scale $\alpha = 5$. Domain-Adaptive Batch Normalization is not used in MNet V2 since we initialize it with ImageNet-pretrained parameters.

Gaussian Prior Maps. The domain-adaptive Gaussian prior maps are constructed at runtime from learned means and standard deviations. The map with index $i = 1, \ldots, N_G$ is computed as

$$g^{(i)}(x, y) = \gamma \exp\left(-\frac{(x - \mu_x^{(i)})^2}{(\sigma_x^{(i)})^2} - \frac{(y - \mu_y^{(i)})^2}{(\sigma_y^{(i)})^2} \right), \tag{1}$$

where $\gamma = 6$ is a scaling factor since the maps are concatenated with the ReLU6 activations of MNet V2. In this formulation, if the standard deviation $\sigma_{xy}^{(i)}$ is optimized over $\mathbb{R}$, then the resulting variance $(\sigma_{xy}^{(i)})^2$ has the domain $\mathbb{R}_{\geq 0}$, which can lead to division by zero. Prior work which uses non-adaptive prior maps [7] addresses this by clipping $\sigma_{xy}^{(i)}$ to a predefined interval $[a, b]$ with $a > 0$ and clipping $\mu_{xy}^{(i)}$ to an interval around the center of the map. However, these constraints potentially limit the ability to learn the optimal parameters. Here, we propose *unconstrained Gaussian prior maps* by substituting $\sigma_{xy}^{(i)} = e^{\lambda_{xy}^{(i)}}$ and optimizing $\lambda_{xy}^{(i)}$ and $\mu_{xy}^{(i)}$ over $\mathbb{R}$. Moreover, instead of drawing the initial Gaussian parameters from a normal distribution, which results in highly correlated maps, we initialize $N_G = 16$ maps as shown in Figure 3 c), covering a broad range of priors. Finally, previous work usually introduces prior maps at the second to last layer in order to model the static center bias. Here, we concatenate the prior maps with the encoder output before the RNN and decoder, in order to leverage the prior maps in higher-level features.

Bypass-RNN. Modeling video saliency data requires a strategy to extract temporal features, such as an RNN, optical flow or 3D convolutions. However, none of these techniques are generally suitable to process static inputs, whereas our goal is to process images and videos with one model. Therefore, we introduce a *Bypass-RNN*, *i.e.*, a RNN whose output is added to its input features via a residual connection that is automatically omitted (bypassed) for static batches. during training and inference. Thus, the RNN only models the residual variations in visual saliency that are caused by temporal features.

Table 1. Network modules and corresponding operations. *ConvDW*(c) denotes a depthwise separable convolution with c channels and kernel size 3×3, followed by batch normalization and ReLU6 activation. *ConvPW*$(c_{\text{in}}, c_{\text{out}})$ is a pointwise 1×1 convolution with c_{in} input and c_{out} output channels, followed by batch normalization and, if $c_{\text{in}} \leq c_{\text{out}}$, by ReLU6 activation. DO(p) denotes 2D dropout with probability p. *Up*(c, n) denotes n-fold upsampling with bilinear interpolation of feature maps with c channels.

Module	Operations
Post-CNN	*ConvDW*(1280), *ConvPW*(1280, 256)
Skip-4x	*ConvPW*(64, 128), DO(0.6), *ConvPW*(128, 64)
Skip-2x	*ConvPW*(160, 256), DO(0.6), *ConvPW*(256, 128)
US1	*Bilinear*(256, 2)
US2	*ConvPW*(384, 768), *ConvDW*(768), *ConvPW*(768, 128), *Up*(128, 2)
Post-US2	*ConvPW*(200, 400), *ConvDW*(400), *ConvPW*(400, 64)
Fusion	*ConvPW(64, 1)*

In the UNISAL model, the *Bypass-RNN* is preceded by a *post-CNN* module, which compresses the concatenated MNet V2 outputs and Gaussian prior maps to 256 channels. For the Bypass-RNN, we use a convolutional GRU (*cGRU*) RNN [44] due to its relative simplicity, followed by a pointwise convolution. The cGRU has 256 hidden channels, 3×3 kernel size, recurrent dropout [9] with probability $p = 0.2$, and MobileNet-style convolutions, *i.e.*, depthwise separable convolutions followed by pointwise convolutions.

Decoder Network and Smoothing. The details of the decoder modules are listed in Table 1. First, the Bypass-RNN features are upsampled to scale $\alpha = 4$ by *US1* and concatenated with the output of *Skip-2x*. Next, the concatenated feature maps are upsampled to scale $\alpha = 3$ by *US2* and concatenated with the output of *Skip-4x*. The *Post-US2* features are reduced to a single channel by an *Domain-Adaptive Fusion* layer (1 × 1 convolution) and upsampled to the input resolution via nearest-neighbor interpolation. The upsampling is followed by a *Domain-Adaptive Smoothing* layer with 41×41 convolutional kernels that explicitly models the dataset-dependent blurring of the ground-truth saliency maps. Finally, following Jetley *et al.* [18], we transform the output into a generalized Bernoulli distribution by applying a softmax operation across all output values.

3.3 Domain-Aware Optimization

Domain-Adaptive Input Resolution. The images/frames have different aspect ratios for each dataset, specifically 4:3 for SALICON, 16:9 for DHF1K, 1.85:1 (median) for Hollywood-2, and 3:2 (median) for UCF Sports. Our network architecture is fully-convolutional, and therefore agnostic to exact the input resolution. Moreover, each mini-batch is constructed from one dataset due to DABN. Therefore, we use input resolutions of 288×384, 224×384, 224×416 and 256×384 for SALICON, DHF1K, Hollywood-2 and UCF Sports, respectively.

Assimilated Frame Rate. The frame rate of the DHF1K videos is 30 fps compared to 24 fps for Hollywood-2 and UCF Sports. In order to assimilate the frame rates during training, and to train on longer time intervals, we construct clips using every 5th frame for DHF1K and every 4th frame for all others, yielding 6 fps overall. During inference, the predictions are interleaved.

4 Experiments

In this section, we compare the proposed method with current state-of-the-art image and video saliency models and provide detailed analyses are presented to gain an understanding of the proposed approach.

4.1 Experimental Setup

Datasets and Evaluation Metrics. To evaluate our proposed unified image and video saliency modeling framework, we jointly train UNISAL on datasets from both modalities. For fair comparison, we use the same training data as [47], i.e., the SALICON [20] image saliency dataset and the Hollywood-2 [34], UCF Sports [34], and DHF1K [47] video saliency datasets. For SALICON, we use the official training/validation/testing split of 10,000/5,000/5,000. For Hollywood-2 and UCF Sports, we use the training and testing splits of 823/884 and 103/47 videos, and the corresponding validation sets are randomly sampled 10% from the training sets, following [47]. Hollywood-2 videos are divided into individual shots. For DHF1K, we use the official training/validation/testing splits of 600/100/300 videos. We compare against the state-of-the-art methods listed in [47] and add newer models with available implementations [35,25,29,7,50]. Moreover, test on the MIT300 benchmark [21], after fine-tuning with the MIT1003 dataset as suggested by the benchmark authors. As in prior work [3,47], we use the evaluation metrics AUC-Judd (AUC-J), Similarity Metric (SIM), shuffled AUC (s-AUC), Linear Correlation Coefficient (CC), and Normalized Scanpath Saliency (NSS) [5].

Implementation Details. We optimize the network via Stochastic Gradient Descent with momentum of 0.9 and weight decay of 10^{-4}. Gradients are clipped to ±2. The learning rate is set to 0.04 and exponentially decayed by a factor of 0.8 after each epoch. The batch size is set to 4 for video data and 32 for SALICON. The video clip length is set to 12 frames that are sampled as described in Section 3.3. Videos that are too short are discarded for training, which applies to Hollywood-2. For comparability, we use the same loss formulation as Wang *et al.* [48]. The model is trained for 16 epochs and with early stopping on the DHF1K validation set. To prevent overfitting, the weights of MNet V2 are frozen for the first two epochs and afterwards trained with a learning rate that is reduced by a factor of 10. The pretrained BN statistics of MNet V2 are frozen throughout training. To account for dataset imbalance, the learning rate for SALICON batches is reduced by a factor of 2. Our model is implemented using the PyTorch framework and trained on a NVIDIA GTX 1080 Ti GPU.

Table 2. Quantitative performance on the video saliency datasets. The training settings (i) to (vi) denote training with: (i) DHF1K, (ii) Hollywood-2, (iii) UCF Sports, (iv) SALICON, (v) DHF1K+Hollywood-2+UCF Sports, and (vi) DHF1K+Hollywood-2+UCF Sports+SALICON. Best performance is shown in **bold** while the second best is underlined. The * symbol denotes training under setting (vi), while † indicates that the method is fine-tuned for each dataset.

	Dataset / Method	DHF1K					Hollywood-2					UCF Sports				
		AUC-J	SIM	s-AUC	CC	NSS	AUC-J	SIM	s-AUC	CC	NSS	AUC-J	SIM	s-AUC	CC	NSS
Dynamic models	PQFT [12]	0.699	0.139	0.562	0.137	0.749	0.723	0.201	0.621	0.153	0.755	0.825	0.250	0.722	0.338	1.780
	Seo *et al.* [41]	0.635	0.142	0.499	0.070	0.334	0.652	0.155	0.530	0.076	0.346	0.831	0.308	0.666	0.336	1.690
	Rudoy *et al.* [39]	0.769	0.214	0.501	0.285	1.498	0.783	0.315	0.536	0.302	1.570	0.763	0.271	0.637	0.344	1.619
	Hou *et al.* [15]	0.726	0.167	0.545	0.150	0.847	0.731	0.202	0.580	0.146	0.684	0.819	0.276	0.674	0.292	1.399
	Fang *et al.* [8]	0.819	0.198	0.537	0.273	1.539	0.859	0.272	0.659	0.358	1.667	0.845	0.307	0.674	0.395	1.787
	OBDL [14]	0.638	0.171	0.500	0.117	0.495	0.640	0.170	0.541	0.106	0.462	0.759	0.193	0.634	0.234	1.382
	AWS-D [27]	0.703	0.157	0.513	0.174	0.940	0.694	0.175	0.637	0.146	0.742	0.823	0.228	0.750	0.306	1.631
	OM-CNN [19]	0.856	0.256	0.583	0.344	1.911	0.887	0.356	0.693	0.446	2.313	0.870	0.321	0.691	0.405	2.089
	Two-stream [1]	0.834	0.197	0.581	0.325	1.632	0.863	0.276	0.710	0.382	1.748	0.832	0.264	0.685	0.343	1.753
	*ACLNet [48]	0.890	0.315	0.601	0.434	2.354	0.913	0.542	0.757	0.623	3.086	0.897	0.406	0.744	0.510	2.567
	TASED-Net [35]	0.895	0.361	0.712	0.470	2.667	0.918	0.507	0.768	0.646	3.302	0.899	0.469	0.752	0.582	2.920
	STRA-Net [25]	0.895	0.355	0.663	0.458	2.558	0.923	0.536	0.774	0.662	3.478	0.910	0.479	0.751	0.593	3.018
	†SalEMA [29]	0.890	**0.465**	0.667	0.449	2.573	0.919	0.487	0.708	0.613	3.186	0.906	0.431	0.740	0.544	2.638
	*SalEMA [29]	0.895	0.283	**0.739**	0.414	2.285	0.875	0.371	0.663	0.456	2.214	0.899	0.381	0.769	0.521	2.503
Static models	ITTI [17]	0.774	0.162	0.553	0.233	1.207	0.788	0.221	0.607	0.257	1.076	0.847	0.251	0.725	0.356	1.640
	GBVS [13]	0.828	0.186	0.554	0.283	1.474	0.837	0.257	0.633	0.308	1.336	0.859	0.274	0.697	0.396	1.818
	SALICON [16]	0.857	0.232	0.590	0.327	1.901	0.856	0.321	0.711	0.425	2.013	0.848	0.304	0.738	0.375	1.838
	Shallow-Net [37]	0.833	0.182	0.529	0.295	1.509	0.851	0.276	0.694	0.423	1.680	0.846	0.276	0.691	0.382	1.789
	Deep-Net [37]	0.855	0.201	0.592	0.331	1.775	0.884	0.300	0.736	0.451	2.066	0.861	0.282	0.719	0.414	1.903
	*Deep-Net [37]	0.874	0.288	0.610	0.374	1.983	0.901	0.482	0.740	0.597	2.834	0.880	0.365	0.729	0.475	2.448
	DVA [46]	0.860	0.262	0.595	0.358	2.013	0.886	0.372	0.727	0.482	2.459	0.872	0.339	0.725	0.439	2.311
	*DVA [46]	0.883	0.297	0.623	0.397	2.237	0.907	0.497	0.753	0.607	2.942	0.892	0.387	0.740	0.492	2.503
	SalGAN [36]	0.866	0.262	0.709	0.370	2.043	0.901	0.393	**0.789**	0.535	2.542	0.876	0.332	0.762	0.470	2.238
UNISAL (ours)	Training setting (i)	0.899	0.378	0.686	0.481	2.707	0.920	0.496	0.710	0.612	3.279	0.896	0.443	0.717	0.553	2.689
	Training setting (ii)	0.881	0.313	0.690	0.422	2.352	0.932	0.534	0.762	0.672	3.803	0.892	0.440	0.735	0.566	2.768
	Training setting (iii)	0.869	0.286	0.664	0.375	2.056	0.890	0.392	0.683	0.475	2.350	0.908	0.502	0.764	0.614	3.076
	Training setting (iv)	0.883	0.288	0.715	0.410	2.259	0.912	0.432	0.750	0.565	2.897	0.892	0.428	0.776	0.561	2.740
	Training setting (v)	**0.901**	0.384	0.692	0.488	2.739	**0.934**	**0.544**	0.758	**0.675**	**3.909**	0.917	0.514	**0.786**	0.642	3.260
	Training setting (vi)	**0.901**	0.390	0.691	**0.490**	**2.776**	**0.934**	0.542	0.759	0.673	3.901	**0.918**	**0.523**	0.775	**0.644**	**3.381**

4.2 Quantitative Evaluation

The results of the quantitative evaluation are shown in Table 2 for the video saliency datasets and in Tables 3 and 4 for the image datasets. For video saliency prediction, in order to analyze the impact of—and generalization across—different datasets, we evaluate six training settings: i) DHF1K, ii) Hollywood-2, iii) UCF Sports, iv) SALICON, v) DHF1K, Hollywood-2, and UCF Sports, vi) DHF1K, Hollywood-2, UCF Sports and SALICON. For fair comparison, we include state-of-the-art methods that are trained on our best-performing training setting (iv): The ACLNet [48] video saliency model and the Deep-Net [37] and DVA [46] image saliency models. In addition, we provide the performance of SalEMA [29], which is based on SalGAN [36], after fine-tuning the model with training setting (vi). Other state-of-the-art video saliency models [19,35,25] are not suitable for training with image data as discussed in Section 1. We observe that the proposed UNISAL model significantly outperforms previous static and dynamic methods, across almost all metrics. We obtain the following additional findings: 1) Training with all video saliency datasets (setting (v)) *always* improves performance compared to individual video saliency datasets (settings (i) to (iii)). This has not

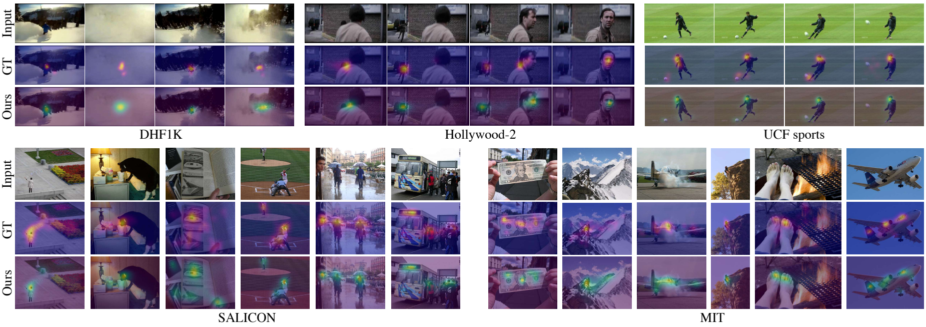

Fig. 4. Qualitative performance of the proposed approach on video (top part) and image (bottom part) saliency prediction.

Table 3. performance on the SALICON and MIT300 benchmarks. Best performance is shown in **bold** while the second best is underlined. Training setting (vi) is used for UNISAL (see supplementary material for other settings).

Dataset / Method	SALICON					MIT300				
	AUC-J	SIM	s-AUC	CC	NSS	AUC-J	SIM	s-AUC	CC	NSS
ITTI [17]	0.667	0.378	0.610	0.205	-	0.75	0.44	0.63	0.37	0.97
GBVS [13]	0.790	0.446	0.630	0.421	-	0.81	0.48	0.63	0.48	1.24
SALICON [16]	-	-	-	-	-	**0.87**	0.60	**0.74**	0.74	2.12
Shallow-Net [37]	0.836	0.520	0.670	0.596	-	0.80	0.46	0.64	0.53	-
Deep-Net [37]	-	-	0.724	0.609	1.859	0.83	0.52	0.69	0.58	1.51
SAM-ResNet [7]	**0.883**	-	0.779	0.842	3.204	**0.87**	**0.68**	0.70	**0.78**	**2.34**
DVA [46]	-	-	-	-	-	0.85	0.58	0.71	0.68	1.98
DINet [50]	**0.884**	-	**0.782**	0.860	**3.249**	0.86	-	0.71	**0.79**	**2.33**
SalGAN [36]	-	-	0.772	0.781	2.459	0.86	0.63	0.72	0.73	2.04
UNISAL (ours)	0.864	**0.775**	0.739	**0.879**	1.952	**0.872**	**0.674**	**0.743**	**0.784**	**2.322**

Table 4. Comparison for dynamic models on the static SALICON benchmark. Best performance is shown in **bold** while the second best is underlined. Training setting (vi) is used for all methods.

Method	AUC-J	SIM	s-AUC	CC	NSS
SalEMA [29]	0.732	0.470	0.519	0.411	0.760
ACLNet [48]	0.843	0.688	0.698	0.771	1.618
UNISAL (w/o DA)	0.848	0.690	0.676	0.799	1.654
UNISAL (final)	**0.864**	**0.775**	**0.739**	**0.879**	**1.952**

been the case for UCF Sports in a previous cross-dataset evaluation study [48]. 2) Additionally including image saliency data (setting (vi)) further improves performance for most metrics for DHF1K and UCF Sports. The exception is Hollywood-2, but the performance decrease is less than 1%.

For image saliency prediction, UNISAL performs on par with state-of-the-art image saliency models both on the SALICON and MIT300 benchmark as shown in Table 3. In addition, we evaluate state-of-the-art video saliency models on SALICON dataset as shown in Table 4. For ACLNet [48] we use the auxiliary output which is trained on SALICON (using the LSTM output yielded worse performance). For SalEMA [29], we fine-tuned their best performing model with training setting (vi). A large performance jump can be observed for the domain-adaptive UNISAL model.

4.3 Qualitative Evaluation

In Figure 4, we show randomly selected saliency predictions for both images and videos. It is visible that the proposed unified model performs well on both

Table 5. Ablation study of the proposed approach on the DHF1K and SALICON validation sets. The proposed components are added incrementally to the baseline to quantify their contribution. Training setting (vi) is used for this study.

Dataset / Config.	DHF1K						SALICON					
	KLD ↓	AUC-J ↑	SIM ↑	s-AUC ↑	CC ↑	NSS ↑	KLD ↓	AUC-J ↑	SIM ↑	s-AUC ↑	CC ↑	NSS ↑
Baseline	1.877	0.863	0.282	0.659	0.372	2.057	0.551	0.824	0.607	0.633	0.711	1.415
+ Gaussian	1.776	0.879	0.300	0.668	0.411	2.273	0.394	0.848	0.675	0.685	0.801	1.634
+ RNNRes	1.754	0.881	0.302	0.666	0.411	2.274	0.450	0.843	0.648	0.665	0.770	1.531
+ SkipConnect	1.749	0.884	0.308	0.658	0.412	2.301	0.404	0.841	0.673	0.664	0.777	1.600
+ Smoothing	1.770	0.882	0.295	0.677	0.416	2.305	0.369	0.848	0.690	0.676	0.799	1.654
+ DomainAdaptive	1.526	0.907	0.373	0.685	0.482	2.731	0.231	0.867	0.768	0.712	0.877	1.925
Final	1.531	0.907	0.381	0.691	0.487	2.755	0.226	0.867	0.771	0.725	0.880	1.923

modalities. For challenging dynamic scenes with complete occlusion (DHF1K, left), the model correctly memorizes the salient object location, indicating that long-term temporal dependencies are effectively modeled. Moreover, the model correctly predicts shifting observer focus in the presence of multiple salient objects, as evident from the Hollywood-2 and UCF Sports samples. The results on static scenes (bottom part of Figure 4) confirm that the proposed unified model indeed generalizes to static scenes.

4.4 Ablation Study

We analyze the contribution of each proposed component: 1) Gaussian prior maps; 2) RNN residual connection; 3) skip connections; 4) *Smoothing* layer; 5) domain-adaptive operations (incl. Bypass-RNN); and 6) domain-aware optimization. We perform the ablation on the representative DHF1K and SALICON validation sets. The results in Table 5 show that each of the proposed components contributes a considerable performance increase. Overall, the domain-adaptive operations contribute the most, both for DHF1K and SALICON. This indicates that mitigating the domain shift between datasets is a crucial component of UNISAL, confirming our initial studies in Section 3.1. The Gaussian prior maps yield the second largest gain, indicating the effectiveness of their proposed unconstrained optimization and early position in the model.

4.5 Inter-Dataset Domain Shift

Figure 5 shows the retrospective analysis of the four domain-adaptive modules. The DABN estimated means in Figure 5 a) are correlated among video datasets with Pearson correlation coefficients r between 82% to 83%, but not correlated between SALICON and the video datasets ($r < 3\%$). Similarly, the DABN variances are least correlated between SALICON and the video datasets (90% vs 92%). This confirms the shift of the feature distributions between datasets, especially between SALICON and the video data. The domain-adaptive *Fusion* layer weights shown in Figure 5 b) are generally correlated across datasets, with $r > 81\%$. However, as for the DABN, SALICON is the least correlated with the

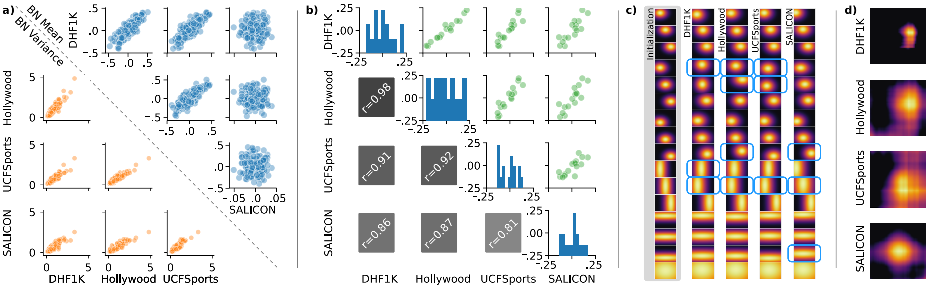

Fig. 5. Retrospective analysis of the domain-adaptive modules. a) Correlation of the batch normalization statistics between datasets (*US2* module, representative). The upper-right plots correlate the estimated means and the lower-left plots the estimated variances. b) Correlation of the *Fusion* layer weights between datasets. The plots on the diagonal show the distribution of weights of the respective dataset. The lower-left part shows Pearson's correlation coefficients. c) Gaussian prior maps. Significant deviations from the initialization are highlighted. d) *Smoothing* kernel of each dataset.

other datasets. Moreover, many of the SALICON *Fusion* weights lie near zero compared to the video datasets, which indicates that only a subset of the video saliency features is relevant for image saliency. The *Domain-Adaptive Fusion* layer models these differences while the remaining network weights are shared. The domain-adaptive Gaussian prior maps shown in Figure 5 c) are successfully learned with our proposed unconstrained parametrization, as observed by the deviations from the initialization. Some prior maps are similar across datasets while others vary visibly, indicating that the different domains have different optimal priors. Finally, the learned *Smoothing* kernels shown in Figure 5 d) vary significantly across datasets. As expected, the DHF1K dataset, which has the least blurry training targets, results in the most narrow *Smoothing* filter.

4.6 Computational Load

With the design of ever more complex network architectures, few studies evaluate the model size, although performance gains can often be traced back to more parameters. We compare the size of UNISAL to the state-of-the-art video saliency predictors in the left column of Table 6. Our model is the most light-weight by a significant margin, with over 5× smaller size than TASED-Net, which is the current state-of-the-art on the DHF1K benchmark (see also Figure 1). The same result applies when comparing to the deep image saliency methods from Table 3, whose sizes range from 92 MB for DVA to 2.5 GB for Shallow-Net.

Another key issue for real-world applications is the model efficiency. Consequently, we present a GPU runtime comparison (processing time per frame) of video saliency models in the right column of Table 6. Our model is the most efficient compared to previous state-of-the-art methods. In addition, we observe a CPU (Intel Xeon W-2123 at 3.60GHz) runtime of 0.43 s (2.3 fps), which is faster than some models' GPU runtime. Considering both the model size and

Table 6. Model size and runtime comparison of saliency prediction methods (based on the DHF1K benchmark [48]). Best performance is shown in **bold**.

Method	Model size (MB)	Method	Runtime (s)
Shallow-Net [37]	2,500	Two-stream [1]	20
STRA-Net [25]	641	SALICON [16]	0.5
SalEMA [29]	364	Shallow-Net [37]	0.1
Two-stream [1]	315	DVA [46]	0.1
ACLNet [48]	250	Deep-Net [37]	0.08
SalGAN [36]	130	TASED-Net [35]	0.06
SALICON [16]	117	ACLNet [48]	0.02
Deep-Net [37]	103	SalGAN [36]	0.02
DVA [46]	96	STRA-Net [25]	0.02
TASED-Net [35]	82	SalEMA [29]	0.01
UNISAL (ours)	**15.5**	UNISAL (ours)	**0.009**

the runtime, the proposed saliency modeling approach achieves state-of-the-art performance in terms of real-world applicability. While the MNet V2 encoder makes a large contribution to low model size and runtime, other contributing factors are: Separable convolutions throughout the cGRU and decoder; cGRU at the low-resolution bottleneck; bilinear upsampling. Without these measures the model size and runtime increase to 59.4 MB and 0.017 s, respectively.

5 Discussion and Conclusion

In this paper, we have presented a simple yet effective approach to unify static and dynamic saliency modeling. To bridge the domain gap, we found it crucial to account for different sources of inter-dataset domain shift through corresponding novel domain-adaptive modules. We integrated the domain-adaptive modules into the new, lightweight and simple UNISAL architecture which is designed to model both data modalities coequally. We observed state-of-the-art performance on video saliency datasets, and competitive performance on image saliency datasets, with a 5 to 20-fold reduction in model size compared to the *smallest* previous deep model, and faster runtime. We found that the domain-adaptive modules capture the differences between image and video saliency data, resulting in improved performance on each individual dataset through joint training. We presented preliminary and retrospective experiments which explain the merit of the domain-adaptive modules. To our knowledge, this is the first attempt towards unifying image and video saliency modeling in a single framework. We believe that our work can serve as a basis for further research into joint modeling of these modalities.

Acknowledgements. We acknowledge the EPSRC (Project Seebibyte, reference EP/M013774/1) and the NVIDIA Corporation for the donation of GPU.

References

1. Bak, C., Kocak, A., Erdem, E., Erdem, A.: Spatio-temporal saliency networks for dynamic saliency prediction. IEEE TMM **20**(7), 1688–1698 (2017)
2. Borji, A.: Saliency Prediction in the Deep Learning Era: An Empirical Investigation. arXiv:1810.03716 (2018)
3. Borji, A., Itti, L.: State-of-the-art in visual attention modeling. IEEE TPAMI **35**(1), 185–207 (2012)
4. Bousmalis, K., Trigeorgis, G., Silberman, N., Krishnan, D., Erhan, D.: Domain separation networks. In: NeurIPS (2016)
5. Bylinskii, Z., Judd, T., Oliva, A., Torralba, A., Durand, F.: What Do Different Evaluation Metrics Tell Us About Saliency Models? IEEE TPAMI **41**(3), 740–757 (2019)
6. Chang, W.G., You, T., Seo, S., Kwak, S., Han, B.: Domain-specific batch normalization for unsupervised domain adaptation. In: CVPR (2019)
7. Cornia, M., Baraldi, L., Serra, G., Cucchiara, R.: Predicting Human Eye Fixations via an LSTM-based Saliency Attentive Model. IEEE TIP **27**(10), 5142–5154 (2016)
8. Fang, Y., Wang, Z., Lin, W., Fang, Z.: Video saliency incorporating spatiotemporal cues and uncertainty weighting. IEEE TIP **23**(9), 3910–3921 (2014)
9. Gal, Y., Ghahramani, Z.: A Theoretically Grounded Application of Dropout in Recurrent Neural Networks. In: NeurIPS (2016)
10. Gorji, S., Clark, J.J.: Going from image to video saliency: Augmenting image salience with dynamic attentional push. In: CVPR (2018)
11. Guo, C., Ma, Q., Zhang, L.: Spatio-temporal saliency detection using phase spectrum of quaternion fourier transform. In: CVPR (2008)
12. Guo, C., Zhang, L.: A novel multiresolution spatiotemporal saliency detection model and its applications in image and video compression. IEEE TIP **19**(1), 185–198 (2009)
13. Harel, J., Koch, C., Perona, P.: Graph-based visual saliency. In: NeurIPS (2007)
14. Hossein Khatoonabadi, S., Vasconcelos, N., Bajic, I.V., Shan, Y.: How many bits does it take for a stimulus to be salient? In: CVPR (2015)
15. Hou, X., Zhang, L.: Dynamic visual attention: Searching for coding length increments. In: NeurIPS (2009)
16. Huang, X., Shen, C., Boix, X., Zhao, Q.: Salicon: Reducing the semantic gap in saliency prediction by adapting deep neural networks. In: ICCV (2015)
17. Itti, L., Koch, C., Niebur, E.: A model of saliency-based visual attention for rapid scene analysis. IEEE TPAMI **20**(11), 1254–1259 (1998)
18. Jetley, S., Murray, N., Vig, E.: End-to-End Saliency Mapping via Probability Distribution Prediction. In: CVPR (2016)
19. Jiang, L., Xu, M., Liu, T., Qiao, M., Wang, Z.: DeepVS: A Deep Learning Based Video Saliency Prediction Approach. In: ECCV (2018)
20. Jiang, M., Huang, S., Duan, J., Zhao, Q.: Salicon: Saliency in context. In: CVPR (2015)
21. Judd, T., Durand, F., Torralba, A.: A Benchmark of Computational Models of Saliency to Predict Human Fixations. Mit-Csail-Tr-2012 **1**, 1–7 (2012)
22. Judd, T., Ehinger, K., Durand, F., Torralba, A.: Learning to predict where humans look. In: ICCV (2009)
23. Kruthiventi, S.S.S., Ayush, K., Babu, R.V.: DeepFix: A Fully Convolutional Neural Network for predicting Human Eye Fixations. IEEE TIP **26**(9), 4446–4456 (2015)

24. Kümmerer, M., Wallis, T.S.A., Bethge, M.: DeepGaze II: Reading fixations from deep features trained on object recognition. arXiv:1610.01563 (2016)
25. Lai, Q., Wang, W., Sun, H., Shen, J.: Video saliency prediction using spatiotemporal residual attentive networks. IEEE TIP (2019)
26. Le Meur, O., Le Callet, P., Barba, D., Thoreau, D.: A coherent computational approach to model bottom-up visual attention. IEEE TPAMI **28**(5), 802–817 (2006)
27. Leboran, V., Garcia-Diaz, A., Fdez-Vidal, X.R., Pardo, X.M.: Dynamic whitening saliency. IEEE TPAMI **39**(5), 893–907 (2016)
28. Li, Y., Wang, N., Shi, J., Liu, J., Hou, X.: Revisiting Batch Normalization For Practical Domain Adaptation. In: ICLR (2016)
29. Linardos, P., Mohedano, E., Nieto, J.J., McGuinness, K., Giro-i Nieto, X., O'Connor, N.E.: Simple vs complex temporal recurrences for video saliency prediction. In: BMVC (2019)
30. Liu, J., Shahroudy, A., Xu, D., Wang, G.: Spatio-temporal lstm with trust gates for 3d human action recognition. In: ECCV (2016)
31. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(Nov), 2579–2605 (2008)
32. Mahadevan, V., Vasconcelos, N.: Spatiotemporal saliency in dynamic scenes. IEEE TPAMI **32**(1), 171–177 (2009)
33. Marat, S., Phuoc, T.H., Granjon, L., Guyader, N., Pellerin, D., Guérin-Dugué, A.: Modelling spatio-temporal saliency to predict gaze direction for short videos. International journal of computer vision **82**(3), 231 (2009)
34. Mathe, Stefan abd Sminchisescu, C.: Actions in the eye: Dynamic gaze datasets and learnt saliency models for visual recognition. IEEE TPAMI **37** (2015)
35. Min, K., Corso, J.J.: Tased-net: Temporally-aggregating spatial encoder-decoder network for video saliency detection. In: ICCV (2019)
36. Pan, J., Ferrer, C.C., McGuinness, K., O'Connor, N.E., Torres, J., Sayrol, E., Giro-i Nieto, X.: Salgan: Visual saliency prediction with generative adversarial networks. arXiv:1701.01081 (2017)
37. Pan, J., Sayrol, E., Giro-i Nieto, X., McGuinness, K., O'Connor, N.E.: Shallow and deep convolutional networks for saliency prediction. In: CVPR (2016)
38. Rozantsev, A., Salzmann, M., Fua, P.: Beyond Sharing Weights for Deep Domain Adaptation. IEEE TPAMI **41**(4), 801–814 (2019)
39. Rudoy, D., Goldman, D.B., Shechtman, E., Zelnik-Manor, L.: Learning video saliency from human gaze using candidate selection. In: CVPR (2013)
40. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: CVPR (2018)
41. Seo, H.J., Milanfar, P.: Static and space-time visual saliency detection by self-resemblance. Journal of vision **9**(12), 15–15 (2009)
42. Sun, Y., Fisher, R.: Object-based visual attention for computer vision. Artificial intelligence **146**(1), 77–123 (2003)
43. Tsai, J.C., Chien, J.T.: Adversarial domain separation and adaptation. In: 2017 IEEE 27th International Workshop on Machine Learning for Signal Processing (MLSP). pp. 1–6 (2017)
44. Valipour, S., Siam, M., Jagersand, M., Ray, N.: Recurrent fully convolutional networks for video segmentation. In: IEEE WACV. pp. 29–36 (2017)
45. Vig, E., Dorr, M., Cox, D.: Large-scale optimization of hierarchical features for saliency prediction in natural images. In: CVPR (2014)
46. Wang, W., Shen, J.: Deep visual attention prediction. IEEE TIP **27**(5), 2368–2378 (2017)

47. Wang, W., Shen, J., Guo, F., Cheng, M.M., Borji, A.: Revisiting video saliency: A large-scale benchmark and a new model. In: CVPR (2018)
48. Wang, W., Shen, J., Xie, J., Cheng, M.M., Ling, H., Borji, A.: Revisiting video saliency prediction in the deep learning era. IEEE TPAMI (2019)
49. Xiao, T., Li, H., Ouyang, W., Wang, X.: Learning Deep Feature Representations with Domain Guided Dropout for Person Re-identification. arXiv:1604.07528 (2016)
50. Yang, S., Lin, G., Jiang, Q., Lin, W.: A dilated inception network for visual saliency prediction. IEEE TMM (2019)
51. Zheng, Q., Jiao, J., Cao, Y., Lau, R.W.: Task-driven webpage saliency. In: ECCV (2018)
52. Zhong, S.h., Liu, Y., Ren, F., Zhang, J., Ren, T.: Video saliency detection via dynamic consistent spatio-temporal attention modelling. In: AAAI (2013)