# Attention Mechanisms in Computer Vision: A Survey

(green mean equation,yellow is important,red is important part in yellow, light blue is standard information)

this survey provides a comprehensive review of various attention mechanisms in computer vision and categorize them according to approach

approaches:

1. channel attention
2. spatial attention
3. temporal attention
4. branch attention

Attention mechanisms can be categorized like figure1 and 2:

1. Channel Attention
2. Channel&Spatial Attention
3. Channel&Spatial Attention
4. Spatial&Temporal Attention
5. Temporal Attention
6. Branch Attention

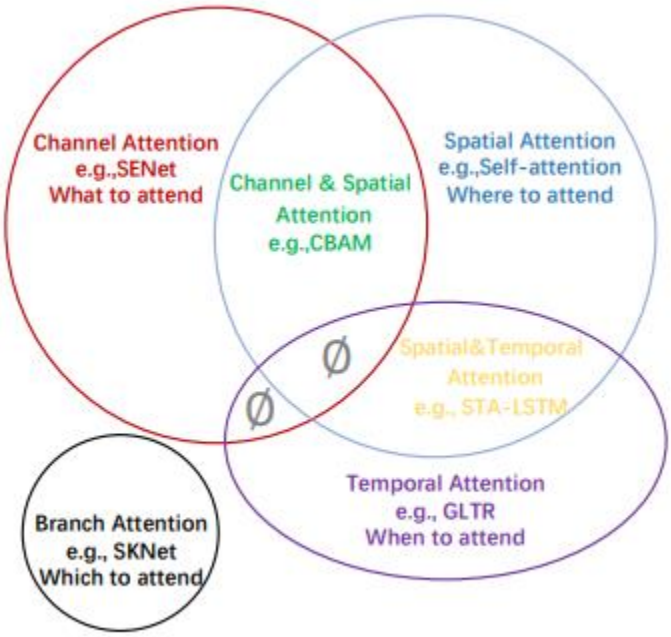



*Figure 1*

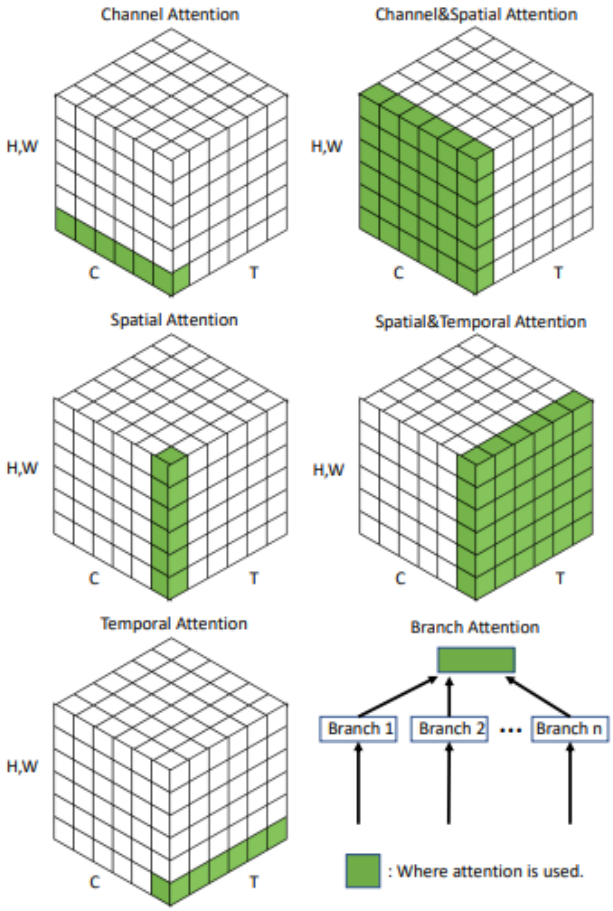



*Figure 2*

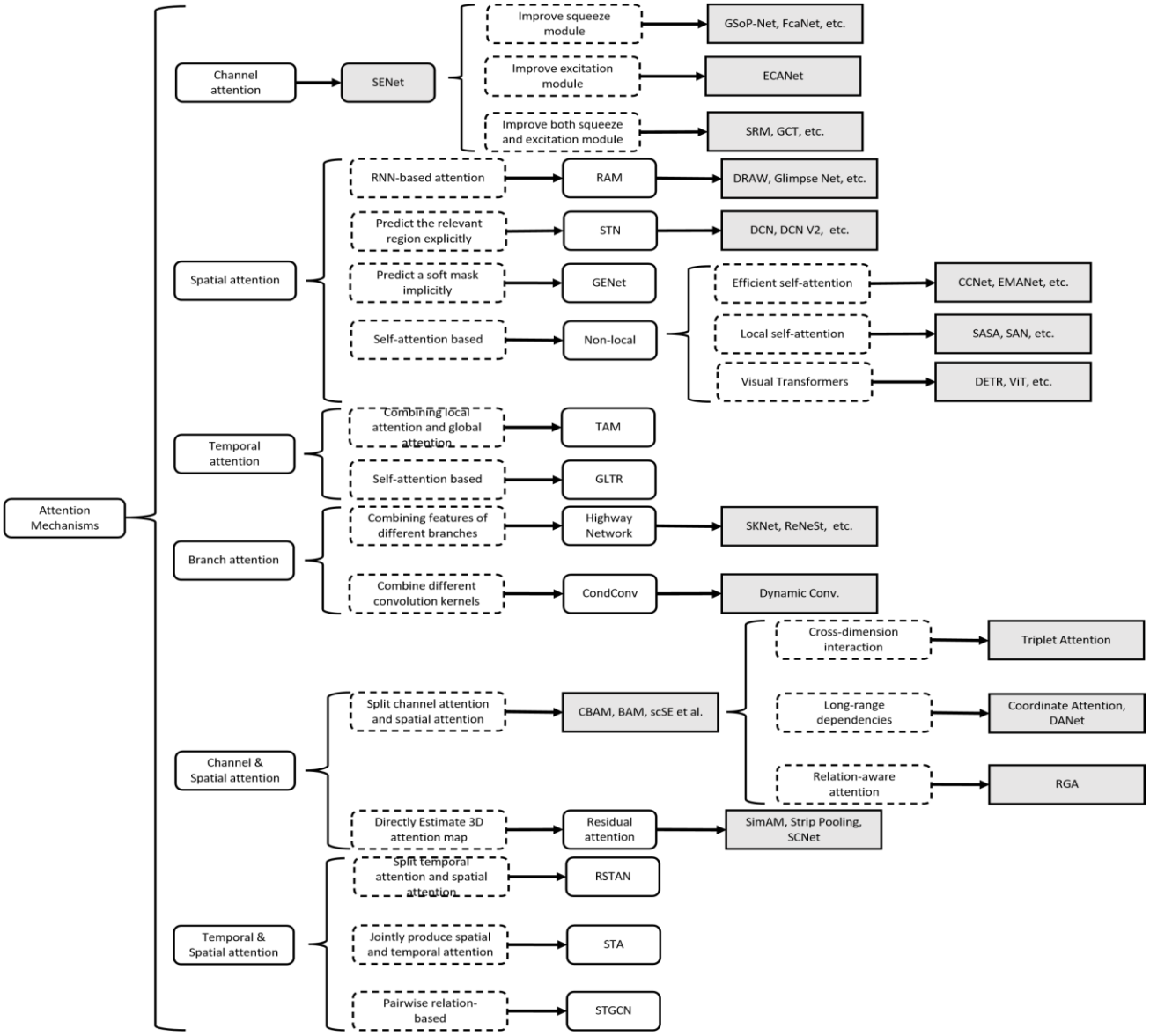
Fig. 1. Ø means such combinations do not (yet) exist.

Fig. 2. Channel, spatial and temporal attention can be regarded as operating on different domains. C represents the channel domain, H and W represent spatial domains, and T means the temporal domain. Branch attention is complementary to these.

| Attention category | Description |
|---|---|
| Channel attention | Generate attention mask across the channel domain and use it to select important channels. |
| Spatial attention | Generate attention mask across spatial domains and use it to select important spatial or predict the most relevant spatial position directly |
| Temporal attention | Generate attention mask in time and use it to select key frames. |
| Branch attention | Generate attention mask across the different branches and use it to select important branches. |
| Channel & spatial attention | Predict channel and spatial attention masks separately or generate a joint 3-D channel, height, width attention mask directly and use it to select important features. |
| Spatial & temporal attention | Compute temporal and spatial attention masks separately or produce a joint spatiotemporal attention mask to focus on informative regions. |

Brief summary of attention categories

Developmental context of visual attention.:




ATTENTION METHODS IN COMPUTER VISION:

1. **General form**: When seeing a scene in our daily life, we will focus on the discriminative regions, and process these regions quickly. The above process can be formulated as

   ***Attention = f(g(x),x)***

   Here g(x) can represent to generate attention which corresponds to the process of attending to the discriminative regions. f(g(x), x) means processing input x based on the attention g(x) which is consistent with processing critical regions and getting information. With the above definition, we find that almost all existing attention mechanisms can be written into the above formulation.
   for example self-attention $g(x)$ and $f(g(x),x)$ can be written as:

   i. $Q,K,V$ = Linear($x$)
   ii. $g(x)$ = Softmax($QK$)
   iii. f(g(x),x) = g(x)V
   iv. For SE, $g(x)$ and $f(g(x),x)$ can be written as:
   $g(x)$ = Sigmoid(MLP(GAP($x$)))
   v. $f(g(x),x) = g(x)x$

2. **Channel Attention:** In deep neural networks, different channels in different feature maps usually represent different objects. Channel attention adaptively recalibrates the weight of each channel, and can be viewed as an object selection process, thus determining *what to pay attention to* first proposed the concept of channel attention and presented SENet for this purpose.

   Summarization of the representative channel attention works and specify process $g(x)$ and $f(g(x),x)$:

   a. **SENet**:

      SENet pioneered channel attention. The core of SENet is a *squeeze-and-excitation* (SE) block which is used to collect global information, capture channel-wise relationships and improve representation ability.

      SE blocks are divided into two parts, a squeeze module and an excitation module. Global spatial information is collected in the squeeze module by global average pooling. The excitation module captures channel-wise relationships and outputs an attention vector by using fully-connected layers and non-linear layers (ReLU and sigmoid). Then, each channel of the input feature is scaled by multiplying the corresponding element in the attention vector. Overall, a squeeze-and-excitation block $F_{se}$ (with parameter $\theta$) which takes $X$ as input and outputs $Y$ can be formulated as:

      $$s = F_{se}(X,\theta) = \sigma(W_2\delta(W_1\text{GAP}(X)))$$
      $$Y = sX$$

      SE blocks play the role of emphasizing important channels while suppressing noise. An SE block can be added after each residual unit due to their low computational resource requirements. However, SE blocks have shortcomings. In the squeeze module, global average pooling is too simple to capture complex global information. In the excitation module, fully-connected layers increase the complexity of the model.as later works attempt to improve the outputs of the squeeze module reduce the complexity of the model by improving the excitation module or improve both the squeeze module and the excitation module .

b. **GSoP-Net**: An SE block captures global information by only using global average pooling (first-order statistics), which limits its modeling capability, in particular the ability to capture high-order statistics.

To address this issue, author proposed to improve the squeeze module by using a *global second-order pooling* (GSoP) block to model high-order statistics while gathering global information.

In the excitation module, a GSoP block performs row-wise convolution to maintain structural information and output a vector. Then a fully-connected layer and a sigmoid function are applied to get a $c$-dimensional attention vector. Finally, it multiplies the input features by the attention vector, as in an SE block. A GSoP block can be formulated as:

$$s = F_{\text{gsop}}(X,\theta) = \sigma(W\text{RC}(\text{Cov}(\text{Conv}(X))))$$
$$Y = sX$$

Here, Conv(·) reduces the number of channels, Cov(·) computes the covariance matrix and RC(·) means row-wise convolution.

By using second-order pooling, GSoP blocks have improved the ability to collect global information over the SE block. However, this comes at the cost of additional computation. Thus, a single GSoP block is typically added after several residual blocks.

c. SRM: Motivated by successes in style transfer, author proposed the lightweight *style-based recalibration module* (SRM). SRM combines style transfer with an attention mechanism. Its main contribution is style pooling which utilizes both mean and standard deviation of the input features to improve its capability to capture global information. It also adopts a lightweight *channel-wise fully-connected* (CFC) layer, in place of the original fully-connected layer, to reduce the computational requirements.

Given an input feature map $X \in \mathrm{R}^{C\times H\times W}$, SRM first collects global information by using style pooling (SP(·)) which combines global average pooling and global standard deviation pooling. Then a channel-wise fully connected (CFC(·)) layer (i.e. fully connected per channel), batch normalization BN and sigmoid function $\sigma$ are used to provide the attention vector. Finally, as in an SE block, the input features are multiplied by the attention vector. Overall, an SRM can be written as:

$$s = F_{\text{srm}}(X,\theta) = \sigma(\text{BN}(\text{CFC}(\text{SP}(X))))$$
$$Y = sX$$

The SRM block improves both squeeze and excitation modules, yet can be added after each residual unit like an SE block.

d. GCT Due to the computational demand and number of parameters of the fully connected layer in the excitation module, it is impractical to use an SE block after each convolution layer. Furthermore, using fully connected layers to model channel relationships is an implicit procedure. To overcome the above problems, author propose

the *gated channel transformation* (GCT) to efficiently collect information while explicitly modeling channel-wise relationships.

Unlike previous methods, GCT first collects global information by computing the $l_2$-norm of each channel. Next, a learnable vector $\alpha$ is applied to scale the feature. Then a competition mechanism is adopted by channel normalization to interact between channels. Like other common normalization methods, a learnable scale parameter $\gamma$ and bias $\beta$ are applied to rescale the normalization. However, unlike previous methods, GCT adopts tanh activation to control the attention vector. Finally, it not only multiplies the input by the attention vector but also adds an identity connection.

A GCT block has fewer parameters than an SE block, and as it is lightweight, can be added after each convolutional layer of a CNN.

e. ECANet

To avoid high model complexity, SENet reduces the number of channels. However, this strategy fails to directly model correspondence between weight vectors and inputs, reducing the quality of results. To overcome this drawback, author proposed the *efficient channel attention* (ECA) block which instead uses a 1D convolution to determine the interaction between channels, instead of dimensionality reduction.

An ECA block has similar formulation to an SE block including a squeeze module for aggregating global spatial information and an efficient excitation module for modeling cross-channel interaction. Instead of indirect correspondence, an ECA block only considers direct interaction between each channel and its $k$-nearest neighbors to control model complexity. Overall, the formulation of an ECA block is:

$$s = F_{\text{eca}}(X,\theta) = \sigma(\text{Conv1D}(\text{GAP}(X)))$$

$$Y = sX$$

where Conv1D(·) denotes 1D convolution with a kernel of shape $k$ across the channel domain, to model local crosschannel interaction. The parameter $k$ decides the coverage of interaction, and in ECA the kernel size $k$ is adaptively determined from the channel dimensionality $C$ instead of by manual tuning, using cross-validation

Compared to SENet, ECANet has an improved excitation module, and provides an efficient and effective block which can readily be incorporated into various CNNs.

f. FcaNet

Only using global average pooling in the squeeze module limits representational ability. To obtain a more powerful representation ability, author rethought global information captured from the viewpoint of compression and analyzed global average pooling in the frequency domain. They proved that global average pooling is a special case of the discrete cosine transform (DCT) and used this observation to propose a novel *multi-spectral channel attention*.

Given an input feature map $X \in \mathrm{R}^{C\times H\times W}$, multispectral channel attention first splits $X$ into many parts $x^i \in \mathrm{R}^{C_0\times H\times W}$. Then it applies a 2D DCT to each part $x^i$. Note that a 2D DCT can

use pre-processing results to reduce computation. After processing each part, all results are concatenated into a vector. Finally, fully connected layers, ReLU activation and a sigmoid are used to get the attention vector as in an SE block.

This work based on information compression and discrete cosine transforms achieves excellent performance on the classification task.

g. EncNet

Inspired by SENet, author proposed the *context encoding module* (CEM) incorporating *semantic encoding loss* (SE-loss) to model the relationship between scene context and the probabilities of object categories, thus utilizing global scene contextual information for semantic segmentation.

Given an input feature map $X \in \mathrm{R}^{C\times H\times W}$, a CEM first learns $K$ cluster centers $D = \{d_1,\ldots,d_K\}$ and a set of smoothing factors $S = \{s_1,\ldots,s_K\}$ in the training phase. Next, it sums the difference between the local descriptors in the input and the corresponding cluster centers using soft-assignment weights to obtain a permutation-invariant descriptor. Then, it applies aggregation to the descriptors of the $K$ cluster centers instead of concatenation for computational efficiency.

In addition to channel-wise scaling vectors, the compact contextual descriptor $e$ is also applied to compute the SE-loss to regularize training, which improves the segmentation of small objects.

Not only does CEM enhance class-dependent feature maps, but it also forces the network to consider big and small objects equally by incorporating SE-loss. Due to its lightweight architecture, CEM can be applied to various backbones with only low computational overhead.

h. Bilinear Attention

Following GSoP-Net, author claimed that previous attention models only use first-order information and disregard higher-order statistical information. They thus proposed a new *bilinear attention block* (bi-attention) to capture local pairwise feature interactions within each channel, while preserving spatial information.

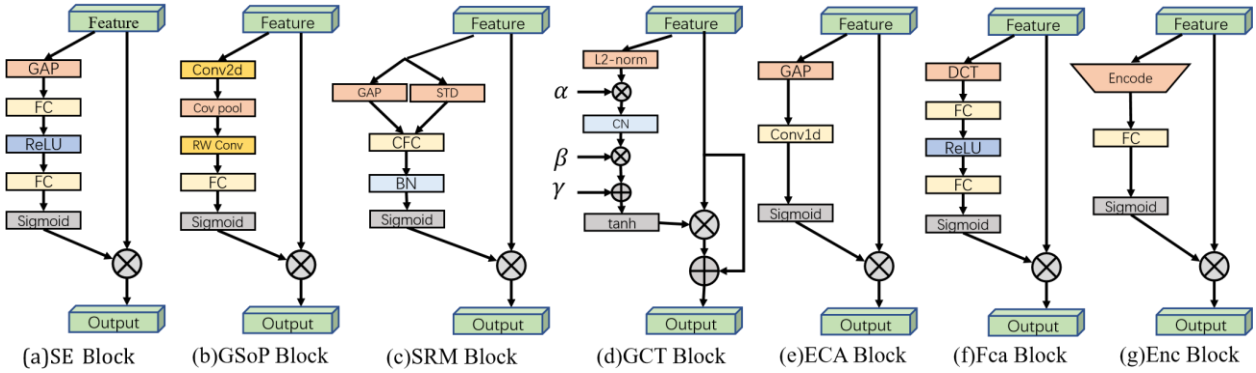
Bi-attention employs the *attention-in-attention* (AiA) mechanism to capture second-order statistical information: the outer point-wise channel attention vectors are computed from the output of the inner channel attention. Formally,

given the input feature map $X$, bi-attention first uses bilinear pooling to capture second-order information $xe = \mathrm{Bi}(\varphi(X)) = \mathrm{Vec}(\mathrm{UTri}(\varphi(X)\varphi(X)^T))$

where $\varphi$ denotes an embedding function used for dimensionality reduction, $\varphi(x)^T$ is the transpose of $\varphi(x)$ across the channel domain, $\mathrm{Utri}(\cdot)$ extracts the upper triangular elements of a matrix and $\mathrm{Vec}(\cdot)$ is vectorization. Then biattention applies the inner

channel attention mechanism to the feature map

The bi-attention block uses bilinear pooling to model the local pairwise feature interactions along each channel, while preserving the spatial information. Using the proposed AiA, the model pays more attention to higher-order statistical information compared with other attention-based models. Bi-attention can be incorporated into any CNN backbone to improve its representational power while suppressing noise.




Various channel attention mechanisms. GAP=global average pooling, GMP=global max pooling, FC=fully-connected layer, Cov pool=Covariance pooling, RW Conv=row-wise convolution, CFC=channel-wise fully connected, CN=channel normalization, DCT=discrete cosine transform.

TABLE

Representative channel attention mechanisms ordered by category and publication date. Their key aims are to emphasize important channels and capture global information. Application areas include: Cls = classification, Det = detection, SSeg = semantic segmentation, ISeg = instance segmentation, ST = style transfer, Action = action recognition.$g(x)$ and $f(g(x),x)$ are the attention process described by Eq. 1. Ranges means the ranges of attention map. S or H means soft or hard attention.(A) channel-wise product. (I) emphasize important channels, (II) capture global information.

| **Category** | **Method** | **Publication** | **Tasks** | $g(x)$ | $f(g(x),x)$ | **Ranges** | **SorH** | **Goals** |
|---|---|---|---|---|---|---|---|---|
| Squeeze-andexcitation network | SENet [5] | CVPR2018 | Cls, Det | global average pooling -> MLP -> sigmoid. | (A) | (0,1) | S | (I),(II) |
| Improve squeeze module | EncNet [53] | CVPR2018 | SSeg | encoder -> MLP -> sigmoid. | (A) | (0,1) | S | (I),(II) |
| | GSoP-Net [54] | CVPR2019 | Cls | 2nd-order pooling -> convolution & MLP -> sigmoid | (A) | (0,1) | S | (I),(II) |
| | FcaNet [57] | ICCV2021 | Cls, Det, ISeg | discrete cosine transform -> MLP -> sigmoid. | (A) | (0,1) | S | (I),(II) |
| Improve excitation module | ECANet [37] | CVPR2020 | Cls, Det, ISeg | global average pooling -> conv1d -> sigmoid. | (A) | (0,1) | S | (I),(II) |
| Improve both squeeze and excitation module | SRM [55] | arXiv2019 | Cls, ST | style pooling -> convolution & MLP -> sigmoid. | (A) | (0,1) | S | (I),(II) |
| | GCT [56] | CVPR2020 | Cls, Det, Action | compute $L2$-norm on spatial -> channel normalization -> tanh. | (A) | (-1,1) | S | (I),(II) |

3. Spatial attention can be seen as an adaptive spatial region selection mechanism: *where to pay attention*.RAM, STN , GENet and Non-Local are representative of different kinds of spatial attention methods. RAM represents RNN-based methods. STN represents those use a sub-network to explicitly predict relevant regions. GENet represents those that use a sub-network implicitly to predict a soft mask to select important regions. Non-Local represents self-attention related methods. In this subsection, we first summarize representative spatial attention mechanisms and specify process $g(x)$ and $f(g(x),x)$

a. RAM

Convolutional neural networks have huge computational costs, especially for large inputs. In order to concentrate limited computing resources on important regions, Mnih proposed the *recurrent attention model* (RAM) that adopts RNNs and reinforcement learning (RL) to make the network learn where to pay attention. RAM pioneered the use of RNNs for visual attention, and was followed by many other RNN-based methods.

b. Glimpse Network

Inspired by how humans perform visual recognition sequentially, Ba proposed a deep recurrent network, similar to RAM, capable of processing a multi-resolution crop of the input image, called a glimpse, for multiple object recognition task. The proposed network updates its hidden state using a glimpse as input, and then predicts a new object as well as the next glimpse location at each step. The glimpse is usually much smaller than the whole image, which makes the network computationally efficient.

the goal of the glimpse network is to extract useful information

Compared to a CNN operating on the entire image, the computational cost of the proposed model is much lower, and it can naturally tackle images of different sizes because it only processes a glimpse in each step. Robustness is additionally improved by the recurrent attention mechanism, which also alleviates the problem of over-fitting. This pipeline can be incorporated into any state-of-the-art CNN backbones or RNN units.

c. Hard and soft attention

To visualize where and what an image caption generation model should focus on, Xu introduced an attention based model as well as two variant attention mechanisms, *hard attention* and *soft attention*.

Given a set of feature vectors $a = \{a_1,\ldots,a_L\}, a_i \in \mathrm{R}^D$ extracted from the input image, the model aims to produce a caption by generating one word at each time step. Thus they adopt a long short-term memory (LSTM) network as a decoder; an attention mechanism is used to generate a contextual vector $z_t$ conditioned on the feature set $a$ and the previous hidden state $h_{t-1}$, where $t$ denotes the time step. Formally, the weight $\alpha_{t,i}$ of the feature vector $a_i$ at the $t$-th time step is defined as

The positive weight $\alpha_{t,i}$ can be interpreted either as the probability that location $i$ is the right place to focus on (hard attention), or as the relative importance of location $i$ to the next word (soft attention). To obtain the contextual vector $z_t$, the hard attention mechanism assigns a multinoulli distribution parametrized

d. Attention Gate

Previous approaches to MR segmentation usually operate on particular regions of interest (ROI), which requires excessive and wasteful use of computational resources and model parameters. To address this issue, Oktay proposed a simple and yet effective mechanism, the *attention gate* (AG), to focus on targeted regions while suppressing feature activations in irrelevant regions.

the attention gate uses additive attention to obtain the gating coefficient. Both the input $X$ and the gating signal are first linearly mapped to an $R^{F\times H\times W}$ dimensional space, and then the output is squeezed in the channel domain to produce a spatial attention weight map $S \in R^{1\times H\times W}$

The attention gate guides the model's attention to important regions while suppressing feature activation in unrelated areas. It substantially enhances the

general and modular, to make it simple to use in various CNN models.

e. STN

The property of translation equivariance makes CNNs suitable for processing image data. However, CNNs lack other transformation invariance such as rotational invariance, scaling invariance and warping invariance. To achieve these attributes while making CNNs focus on important regions, Jaderberg proposed *spatial transformer networks* (STN) that use an explicit procedure to learn invariance to translation, scaling, rotation and other more general warps, making the network pay attention to the most relevant regions. STN was the first attention mechanism to explicitly predict important regions and provide a deep neural network with transformation invariance. Various following works have had even greater success.

Taking a 2D image as an example a 2D affine transformation

After obtaining the correspondence, the network can sample relevant input regions using the correspondence. To ensure that the whole process is differentiable and can be updated in an end-to-end manner, bilinear sampling is used to sample the input features

STNs focus on discriminative regions automatically and learn invariance to some geometric transformations.

f. Deformable Convolutional Networks

With similar purpose to STNs, Dai proposed *deformable convolutional networks* (deformable ConvNets) to be invariant to geometric transformations, but they pay attention to the important regions in a different manner.

Specifically, deformable ConvNets do not learn an affine transformation. They divide convolution into two steps, firstly sampling features on a regular grid $R$ from the input feature map, then aggregating sampled features by weighted summation using a convolution kernel.

To address this problem, bilinear interpolation is used. Deformable RoI pooling is also used, which greatly improves object detection.

this is important in object detection and semantic segmentation tasks.

g. Self-attention and variants

Self-attention was proposed and has had great success in the field of *natural language processing* (NLP). Recently, it has also shown the potential to become a dominant tool in computer vision. Typically, self-attention is used as a spatial attention mechanism to capture global information.

Due to the localization of the convolutional operation,CNNs have inherently narrow receptive fields, which limits the ability of CNNs to understand scenes globally. To increase the receptive field, Wang introduced self-attention into computer vision. Taking a 2D image as an example, given a feature map $F \in \mathrm{R}^{C \times H \times W}$, self-attention first computes the queries, keys and values $Q,K,V \in \mathrm{R}^{C_0 \times N}, N = H \times W$ by linear projection and reshaping operations. Then self-attention can be formulated as:

$A = (a)_{i,j} =$
Softmax$(QK^T)$,
$Y = AV$,

Self-attention is a powerful tool to model global information and is useful in many visual tasks.

However, the self-attention mechanism has several shortcomings, particularly its quadratic complexity, which limit its applicability. Several variants have been introduced to alleviate these problems. The *disentangled non-local* approach improves self-attention's accuracy and effectiveness, but most variants focus on reducing its computational complexity.

CCNet regards the self-attention operation as a graph convolution and replaces the densely-connected graph processed by self-attention with several sparsely-connected graphs. To do so, it proposes *criss-cross attention* which considers row attention and column attention recurrently to obtain global information. CCNet reduces the complexity of self-attention

EMANet views self-attention in terms of expectation maximization (EM). It proposes *EM attention* which adopts the EM algorithm to get a set of compact bases instead of using all points as reconstruction bases. This reduces the complexity from $O(N^2)$ to $O(NK)$, where $K$ is the number of compact bases.

GCNet analyses the attention map used in self-attention and finds that the global contexts obtained by self-attention are similar for different query positions in the same image. Thus, it first proposes to predict a single attention map shared by all query points, and then gets global information from a weighted sum of input features according to this attention map. This is like average pooling, but is a more general process for collecting global information.

$A^2$Net is motivated by SENet to divide attention into feature gathering and feature distribution processes, using two different kinds of attention. The first aggregates global information via second-order attention pooling and the second distributes the global descriptors by soft selection attention.

GloRe understands self-attention from a graph learning perspective. It first collects $N$ input features into $M \ll N$ nodes and then learns an adjacency matrix of global interactions between nodes. Finally, the nodes distribute global information to input features. A similar idea can be found in LatentGNN , MLP-Mixer and ResMLP .

OCRNet proposes the concept of *object-contextual representation* which is a weighted aggregation of all object regions' representations in the same category, such as a weighted average of all car region representations. It replaces the key and vector with this object-contextual representation leading to successful improvements in both speed and effectiveness.

Yin deeply analyzed the self-attention mechanism resulting in the core idea of decoupling self-attention into a pairwise term and a unary term. The pairwise term focuses on modeling relationships while the unary term focuses on salient boundaries. This decomposition prevents unwanted interactions between the two terms, greatly improving semantic segmentation, object detection and action recognition.

HamNet models capturing global relationships as a low-rank completion problem and designs a series of white-box methods to capture global context using matrix decomposition. This not only reduces the complexity, but increases the interpretability of self-attention.

EANet proposes that self-attention should only consider correlation in a single sample and should ignore potential relationships between different samples. To explore the correlation between different samples and reduce computation, it makes use of an external attention that adopts learnable, lightweight and shared key and value vectors. It further reveals that using softmax to normalize the attention map is not optimal and presents double normalization as a better alternative.

self-attention also can be used to replace convolution operations for aggregating neighborhood information.

SASA suggests that using self-attention to collect global information is too computationally intensive and instead adopts local self-attention to replace all spatial convolution in a CNN. The authors show that doing so improves speed, number of parameters and quality of results. They also explores the behavior of positional embedding and show that relative positional embeddings are suitable. Their work also studies how to combine local self-attention with convolution.

LR-Net appeared concurrently with SASA. It also studies how to model local relationships by using local self-attention. A comprehensive study probed the effects of positional embedding, kernel size, appearance composability and adversarial attacks.

SAN explored two modes, pairwise and patchwise, of utilizing attention for local feature aggregation. It proposed a novel vector attention adaptive both in content and channel, and assessed its effectiveness both theoretically and practically. In addition to providing significant improvements in the image domain, it also has been proven useful in 3D point cloud processing.

h. Vision Transformers

Transformers have had great success in natural language processing .Recently, iGPT and DETR demonstrated the huge potential for transformer-based models in computer vision. Motivated by this, Dosovitskiy proposed the *vision transformer* (ViT) which is the first pure transformer architecture for image processing. It is capable of achieving comparable results to modern convolutional neural networks.

ViT demonstrates that a pure attention-based network can achieve better results than a convolutional neural network especially for large datasets such as JFT-300 and ImageNet-21K .

Following ViT, many transformer-based architectures such as PCT , IPT , T2T-ViT , DeepViT , SETR , PVT , CaiT , TNT , Swintransformer, Query2Label , MoCoV3 , BEiT , SegFormer , FuseFormer and MAE have appeared, with excellent results for many kind of visual tasks including image classification, object detection, semantic segmentation, point cloud processing, action recognition and self-supervised learning.

Inspired by SENet, Hu designed GENet to capture long-range spatial contextual information by providing a recalibration function in the spatial domain.

GENet combines part gathering and excitation operations. In the first step, it aggregates input features over large neighborhoods and models the relationship between different spatial locations. In the second step, it first generates an attention map of the same size as the input feature map, using interpolation. Then each position in the input feature map is scaled by multiplying by the corresponding element in the attention map.

The gather-excite module is lightweight and can be inserted into each residual unit like an SE block. It emphasizes important features while suppressing noise.

PSANet

Motivated by success in capturing long-range dependencies in convolutional neural networks, Zhao presented the novel PSANet framework to aggregate global information. It models information aggregation as an information flow and proposes a bidirectional information propagation mechanism to make information flow globally.

It can be added to the end of a convolutional neural network as an effective complement to greatly improve semantic segmentation.

Representative spatial attention mechanisms sorted by category and date. Application areas include: Cls = classification, FGCls = fine-grained classification, Det = detection, SSeg = semantic segmentation, ISeg = instance segmentation, ST = style transfer, Action = action recognition, ICap = image captioning.$g(x)$ and $f(g(x),x)$ are the attention process described by Eq. 1. Ranges means the ranges of attention map. S or H means soft or hard attention.(A) choose region according to the prediction. (B) element-wise product, (C)aggregate information via attention map. (I)focus thenetwork on discriminative regions, (II) avoid excessive computation for large input images, (III) provide more transformation

invariance, (IV) capture long-range dependencies, (V) denoise input feature map (VI) adaptively aggregate neighborhood information, (VII) reduce inductive bias.

| Category | Method | Publication | Tasks | $g(x)$ | $f(g(x),x)$ | Ranges | SorH | Goals |
|---|---|---|---|---|---|---|---|---|
| RNN-based methods | RAM [31] | NIPS2014 | Cls | use RNN to recurrently predict important re-gions | (A) | (0,1) | H | (I), (II). |
| | Hard and soft attention [35] | ICML2015 | ICap | a)compute similarity between visual features and previous hidden state -> interpret attention weight. | (C) | (0,1) | S, H | (I). |
| Predict the relevant region ex-plictly | STN [32] | NIPS2015 | Cls, FGCls | use sub-network to predict an affine transformation. | (A) | (0,1) | H | (I), (III). |
| | DCN [7] | ICCV2017 | Det, SSeg | use sub-network to predict offset coordinates. | (A) | (0,1) | H | (I), (III). |
| Predict the relevant region im-plictly | GENet [61] | NIPS2018 | Cls, Det | average pooling or depth-wise convolution -> interpolation -> sigmoid | (B) | (0,1) | S | (I). |
| | PSANet [87] | ECCV2018 | SSeg | predict an attention map using a sub-network. | (C) | (0,1) | S | (I), (IV). |
| Self-attention based methods | Non-Local [15] | CVPR2018 | Action, Det, ISeg | Dot product between query and key -> softmax | (C) | (0,1) | S | (I), (IV), (V) |
| | SASA [43] | NeurIPS2019 | Cls, Det | Dot product between query and key -> softmax. | (C) | (0,1) | S | (I), (VI) |
| | ViT [34] | ICLR2021 | Cls | divide the feature map into multiple groups > Dot product between query and key -> softmax. | (C) | (0,1) | S | (I),(IV), (VII). |

## 4 Temporal Attention

Temporal attention can be seen as a dynamic time selection mechanism determining *when to pay attention*, and is thus usually used for video processing. Previous works often emphasise how to capture both short-term and long-term cross-frame feature dependencies.

### *4.1 Self-attention and variants*

RNN and temporal pooling or weight learning have been widely used in work on video representation learning to capture interaction between frames, but these methods have limitations in terms of either efficiency or temporal relation modeling.

To overcome them, Li proposed a *globallocal temporal representation* (GLTR) to exploit multi-scale temporal cues in a video sequence. GLTR consists of a *dilated temporal pyramid* (DTP) for local temporal context learning and a *temporal self attention* module for capturing global temporal interaction. DTP adopts dilated convolution with dilatation rates increasing progressively to cover various temporal ranges, and then concatenates the various outputs to aggregate multi-scale information.

The short-term temporal contextual information from neighboring frames helps to distinguish visually similar regions while the long-term temporal information serves to overcome occlusions and noise. GLTR combines the advantages of both modules, enhancing representation capability and suppressing noise. It can be incorporated into any stateof-the-art CNN backbone to learn a global descriptor for a whole video. However, the self-attention mechanism has quadratic time complexity, limiting its application.

### *4.2 TAM*

To capture complex temporal relationships both efficiently and flexibly, Liu proposed a *temporal adaptive module* (TAM). It adopts an adaptive kernel instead of selfattention to capture global contextual information, with lower time complexity than GLTR .

TAM has two branches, a local branch and a global branch. Given the input feature map $X \in \mathrm{R}^{C\times T\times H\times W}$, global spatial average pooling GAP is first applied to the feature map to ensure TAM has a low computational cost. Then the local branch in TAM employs several 1D convolutions with ReLU nonlinearity across the temporal domain to produce locationsensitive importance maps for enhancing frame-wise features. The local branch can be written as $s = \sigma(\mathrm{Conv1D}(\delta(\mathrm{Conv1D}(\mathrm{GAP}(X)))))$ (61)

$$X^1 = sX.$$

Unlike the local branch, the global branch is location invariant and focuses on generating a channel-wise adaptive kernel based on global temporal information in each channel. For the $c$-th channel, the kernel can be written as

$$\Theta_c = \mathrm{Softmax}(\mathrm{FC}_2(\delta(\mathrm{FC}_1(\mathrm{GAP}(X)_c))))$$

where $\Theta_c \in \mathrm{R}^K$ and $K$ is the adaptive kernel size. Finally, TAM convolves the adaptive kernel $\Theta$ with $X_{\mathrm{out}}{}^1$:

$$Y = \Theta \otimes X^1$$

With the help of the local branch and global branch, TAM can capture the complex temporal structures in video and enhance per-frame features at low computational cost. Due to its flexibility and lightweight design, TAM can be added to any existing 2D CNNs.

Representative temporal attention mechanisms sorted by date. ReID = re-identification, Action = action recognition. Ranges means the ranges of attention map. S or H means soft or hard attention. $g(x)$ and $f(g(x),x)$ are the attention process described by Eq. 1. (A) aggregate information via attention map. (I) exploit multi-scale short-term temporal contextual information (II)capture long-term temporal feature dependencies (III) capture local temporal contexts

| Category | Method | Publication | Tasks | $g(x)$ | $f(g(x),x)$ | Ranges | SorH | Goals |
|---|---|---|---|---|---|---|---|---|
| Self-attention based methods | GLTR [171] | ICCV2019 | ReID | dilated 1D Convs -> selfattention in temporal dimension | (A) | (0,1) | S | (I), (II). |
| Combine local attention and global attention | TAM [172] | Arxiv2020 | Action | a)local: global spatial average pooling -> 1D Convs, b) global: global spatial average pooling -> MLP -> adaptive convolution | (A) | (0,1) | S | (II), (III). |

## 5 Branch Attention

Branch attention can be seen as a dynamic branch selection mechanism: *which to pay attention to*, used with a multi-branch structure.

### 5.1 Highway networks

Inspired by the *long short term memory* network, Srivastava proposed *highway networks* that employ adaptive gating mechanisms to enable information flows across layers to address the problem of training very deep networks.

The gating mechanism and skip-connection structure make it possible to directly train very deep highway networks using simple gradient descent methods. Unlike fixed skip-connections, the gating mechanism adapts to the input, which helps to route information across layers. A highway network can be incorporated in any CNN.

### 5.2 SKNet

Research in the neuroscience community suggests that visual cortical neurons adaptively adjust the sizes of their receptive fields (RFs) according to the input stimulus . This inspired Li to propose an automatic selection operation called *selective kernel* (SK) convolution.

SK convolution is implemented using three operations: split, fuse and select. During split, transformations with different kernel sizes are applied to the feature map to obtain different sized RFs. Information from all branches is then fused together via element-wise summation to compute the gate vector. This is used to control information flows from the multiple branches. Finally, the output feature map is obtained by aggregating feature maps for all branches, guided by the gate vector.

SK convolutions enable the network to adaptively adjust neurons' RF sizes according to the input, giving a notable improvement in results at little computational cost. The gate mechanism in SK convolutions is used to fuse information from multiple branches. Due to its lightweight design, SK convolution can be applied to any CNN backbone by replacing all large kernel convolutions. ResNeSt also adopts this attention mechanism to improve the CNN backbone in a more general way, giving excellent results on ResNet and ResNeXt.

### 5.3 CondConv

A basic assumption in CNNs is that all convolution kernels are the same. Given this, the typical way to enhance the representational power of a network is to increase its depth or width, which introduces significant extra computational cost. In order to more efficiently increase the capacity of convolutional neural networks, Yang proposed a novel multi-branch operator called CondConv.

CondConv makes full use of the advantages of the multibranch structure using a branch attention method with little computing cost. It presents a novel manner to efficiently increase the capability of networks.

### *5.4 Dynamic Convolution*

The extremely low computational cost of lightweight CNNs constrains the depth and width of the networks, further decreasing their representational power. To address the above problem, Chen proposed *dynamic convolution*, a novel operator design that increases representational power with negligible additional computational cost and does not change the width or depth of the network in parallel with CondConv.

Dynamic convolution uses $K$ parallel convolution kernels of the same size and input/output dimensions instead of one kernel per layer. Like SE blocks, it adopts a squeezeand-excitation mechanism to generate the attention weights for the different convolution kernels. These kernels then aggregated dynamically by weighted summation and applied to the input feature map $X$:

$$s = \text{softmax}(W_2\delta(W_1\text{GAP}(X)))$$

$$\text{DyConv} = \sum_{i=1}^{K} s_k \text{Conv}_k$$

$$Y = \mathrm{DyConv}(X)$$

Here the convolutions are combined by summation of weights and biases of convolutional kernels.

Compared to applying convolution to the feature map, the computational cost of squeeze-and-excitation and weighted summation is extremely low. Dynamic convolution thus provides an efficient operation to improve representational power and can be easily used as a replacement for any convolution.

TABLE 6
Representative branch attention mechanisms sorted by date. Cls = classification, Det=Object Detection. $g(x)$ and $f(g(x),x)$ are the attention process described by Eq. 1.Ranges means the ranges of attention map. S or H means soft or hard attention. (A) element-wise product. (B) channel-wise product. (C) aggergate information via attention. (I)overcome the problem of vanishing gradient (II) dynamically fuse different branches. (III) adaptively select a suitable receptive field (IV) improve the performance of standard convolution (be) dynamically fuse different convolution kernels.

| Category | Method | Publication | Tasks | $g(x)$ | $f(g(x),x)$ | Ranges | SorH | Goals |
|---|---|---|---|---|---|---|---|---|
| Combine different branches | Highway Network [113] | ICML2015W | Cls | linear layer -> sigmoid | (A) | (0,1) | S | (I), (II). |
| | SKNet [114] | CVPR2019 | Cls | global average pooling -> MLP -> softmax | (B) | (0,1) | S | (II), (III) |
| Combine different convolution kernels | CondConv [173] | NeurIPS2019 | Cls, Det | global average pooling > linear layer -> sigmoid | (C) | (0,1) | S | (IV), (V). |

## 6 Channel & Spatial Attention

Channel & spatial attention combines the advantages of channel attention and spatial attention. It adaptively selects both important objects and regions. The *residual attention network* pioneered the field of channel & spatial attention, emphasizing the importance of informative features in both spatial and channel dimensions. It adopts a bottom-up structure consisting of several convolutions to produce a 3D (height, width, channel) attention map. However, it has high computational cost and limited receptive fields.

To leverage global spatial information later works enhance discrimination of features by introducing global average pooling, as well as decoupling channel attention and spatial channel attention for computational efficiency. Other works apply self-attention mechanisms for channel & spatial attention to explore pairwise interaction. Yet further works adopt the spatial-channel attention mechanism to enlarge the receptive field.

### *6.1 Residual Attention Network*

Inspired by the success of ResNet , Wang proposed the very deep convolutional *residual attention network* (RAN) by combining an attention mechanism with residual connections.

Each attention module stacked in a residual attention network can be divided into a mask branch and a trunk branch. The trunk branch processes features, and can be implemented by any state-of-the-art structure including a pre-activation residual unit and an inception block. The mask branch uses a bottom-up top-down structure to learn a mask of the same size that softly weights output features from the trunk branch. A sigmoid layer normalizes the output to $[0,1]$ after two $1 \times 1$ convolution layers.

Inside each attention module, a bottom-up top-down feedforward structure models both spatial and cross-channel dependencies, leading to a consistent performance improvement. Residual attention can be incorporated into any deep network structure in an end-to-end training fashion. However, the proposed bottom-up top-down structure fails to leverage global spatial information. Furthermore, directly predicting a 3D attention map has high computational cost.

### *6.2 CBAM*

To enhance informative channels as well as important regions,Woo proposed the *convolutional block attention module* (CBAM) which stacks channel attention and spatial attention in series. It decouples the channel attention map and spatial attention map for computational efficiency, and leverages spatial global information by introducing global pooling.

CBAM has two sequential sub-modules, channel and spatial. Given an input feature map $X \in \mathrm{R}^{C\times H\times W}$ it sequentially infers a 1D channel attention vector $s_c \in \mathrm{R}^{C}$ and a 2D spatial attention map $s_s \in \mathrm{R}^{H\times W}$. The formulation of the channel attention sub-module is similar to that of an SE block, except that it adopts more than one type of pooling operation to aggregate global information. In detail, it has two parallel branches using max-pool and avg-pool operations

where $\mathrm{GAP}^{s}$ and $\mathrm{GMP}^{s}$ denote global average pooling and global max pooling operations in the spatial domain. The spatial attention sub-module models the spatial relationships of features, and is complementary to channel attention. Unlike channel attention, it applies a convolution layer with a large kernel to generate the attention map

Combining channel attention and spatial attention sequentially, CBAM can utilize both spatial and cross-channel relationships of features to tell the network *what* to focus on and *where* to focus. To be more specific, it emphasizes useful channels as well as enhancing informative local regions. Due to its lightweight design, CBAM can be integrated into any CNN architecture seamlessly with negligible additional cost. Nevertheless, there is still room for improvement in the channel & spatial attention mechanism. For instance, CBAM adopts a convolution to produce the spatial attention map, so the spatial sub-module may suffer from a limited receptive field.

### *6.3 BAM*

At the same time as CBAM, Park proposed the *bottleneck attention module* (BAM), aiming to efficiently improve the representational capability of networks. It uses dilated convolution to enlarge the receptive field of the spatial attention sub-module, and build a *bottleneck structure* as suggested by ResNet to save computational cost.

For a given input feature map $X$, BAM infers the channel attention $s_c \in \mathrm{R}^{C}$ and spatial attention $s_s \in \mathrm{R}^{H\times W}$ in two parallel streams, then sums the two attention maps after resizing both branch outputs to $\mathrm{R}^{C\times H\times W}$. The channel attention branch, like an SE block, applies global average pooling to the feature map to aggregate global information, and then uses an MLP with channel dimensionality reduction. In order to utilize contextual information effectively, the spatial attention branch combines a bottleneck structure and dilated convolutions. Overall,

BAM can emphasize or suppress features in both spatial and channel dimensions, as well as improving the representational power. Dimensional reduction applied to both channel and spatial attention branches enables it to be integrated with any convolutional neural network with little extra computational cost. However, although dilated convolutions enlarge the receptive field effectively, it still fails to capture long-range contextual information as well as encoding crossdomain relationships.

### *6.4 scSE*

To aggregate global spatial information, an SE block applies global pooling to the feature map. However, it ignores pixel-wise spatial information, which is important in dense prediction tasks. Therefore, Roy proposed *spatial and channel SE blocks* (scSE). Like BAM, spatial SE blocks are used, complementing SE blocks, to provide spatial attention weights to focus on important regions.

Given the input feature map $X$, two parallel modules, spatial SE and channel SE, are applied to feature maps to encode spatial and channel information respectively. The channel SE module is an ordinary SE block, while the spatial SE module adopts $1 \times 1$ convolution for spatial squeezing. The outputs from the two modules are fused.

The proposed scSE block combines channel and spatial attention to enhance features as well as capturing pixel-wise spatial information. Segmentation tasks are greatly benefited as a result. The integration of an scSE block in F-CNNs makes a consistent improvement in semantic segmentation at negligible extra cost.

### *6.5 Triplet Attention*

In CBAM and BAM, channel attention and spatial attention are computed independently, ignoring relationships between these two domains. Motivated by spatial attention, Misra proposed *triplet attention*, a lightweight but effective attention mechanism to capture cross-domain interaction.

Given an input feature map $X$, triplet attention uses three branches, each of which plays a role in capturing cross-domain interaction between any two domains from $H$, $W$ and $C$. In each branch, rotation operations along different axes are applied to the input first, and then a Zpool layer is responsible for aggregating information in the zeroth dimension. Finally, a standard convolution layer with kernel size $k \times k$ models the relationship between the last two domains.

Unlike CBAM and BAM, triplet attention stresses the importance of capturing cross-domain interactions instead of computing spatial attention and channel attention independently. This helps to capture rich discriminative feature representations. Due to its simple but efficient structure, triplet attention can be easily added to classical backbone networks.

### *6.6 SimAM*

Yang also stress the importance of learning attention weights that vary across both channel and spatial domains in proposing SimAM, a simple, parameterfree attention module capable of directly estimating 3D weights instead of expanding 1D or 2D weights. The design of SimAM is based on well-known neuroscience theory, thus avoiding need for manual fine tuning of the network structure.

Motivated by the spatial suppression phenomenon , they propose that a neuron which shows suppression effects should be emphasized and define an energy function for each neuron

This work simplifies the process of designing attention and successfully proposes a novel 3-D weight parameter-free attention module based on mathematics and neuroscience theories.

### *6.7 Coordinate attention*

An SE block aggregates global spatial information using global pooling before modeling cross-channel relationships, but neglects the importance of positional information. BAM and CBAM adopt convolutions to capture local relations, but fail to model long-range dependencies. To solve these problems, Hou proposed *coordinate attention*, a novel attention mechanism which embeds positional information into channel attention, so that the network can focus on large important regions at little computational cost. The coordinate attention mechanism has two consecutive steps, coordinate information embedding and coordinate attention generation. First, two spatial extents of pooling kernels encode each channel horizontally and vertically. In the second step, a shared $1 \times 1$ convolutional transformation function is applied to the concatenated outputs of the two pooling layers. Then coordinate attention splits the resulting tensor into two separate tensors to yield attention vectors with the same number of channels for horizontal and vertical coordinates of the input $X$ along.

Using coordinate attention, the network can accurately obtain the position of a targeted object. This approach has a larger receptive field than BAM and CBAM. Like an SE block, it also models cross-channel relationships, effectively enhancing the expressive power of the learned features. Due to its lightweight design and flexibility, it can be easily used in classical building blocks of mobile networks.

### *6.8 DANet*

In the field of scene segmentation, encoder-decoder structures cannot make use of the global relationships between objects, whereas RNN-based structures heavily rely on the output of the long-term memorization. To address the above problems, Fu proposed a novel framework, the *dual attention network* (DANet), for natural scene image segmentation. Unlike CBAM and BAM, it adopts a selfattention mechanism instead of simply stacking convolutions to compute the spatial attention map, which enables the network to capture global information directly.

DANet uses in parallel a position attention module and a channel attention module to capture feature dependencies in spatial and channel domains. Given the input feature map $X$, convolution layers are applied first in the position attention module to obtain new feature maps. Then the position attention module selectively aggregates the features at each position using a weighted sum of features at all positions, where the weights are determined by feature similarity between corresponding pairs of positions. The channel attention module has a similar form except for dimensional reduction to model cross-channel relations. Finally the outputs from the two branches are fused to obtain final feature representations. For simplicity, we reshape the feature map $X$ to $C \times (H \times W)$ whereupon

The position attention module enables DANet to capture long-range contextual information and adaptively integrate similar features at any scale from a global viewpoint, while the channel attention module is responsible for enhancing useful channels as well as suppressing noise. Taking spatial and channel relationships into consideration explicitly improves the feature representation for scene segmentation. However, it is computationally costly, especially for large input feature maps.

### *6.9 RGA*

Unlike coordinate attention and DANet, which emphasise capturing long-range context, in *relation-aware global attention* (RGA) , Zhang stress the importance of global structural information provided by pairwise relations, and uses it to produce attention maps.

RGA uses global relations to generate the attention score for each feature node, so provides valuable structural information and significantly enhances the representational power. RGA-S and RGA-C are

flexible enough to be used in any CNN network; Zhang propose using them jointly in sequence to better capture both spatial and cross-channel relationships.

### *6.10 Self-Calibrated Convolutions*

Motivated by the success of group convolution, Liu presented *self-calibrated convolution* as a means to enlarge the receptive field at each spatial location.

Self-calibrated convolution is used together with a standard convolution. It first divides the input feature $X$ into $X_1$ and $X_2$ in the channel domain. The self-calibrated convolution first uses average pooling to reduce the input size and enlarge the receptive field

Such self-calibrated convolution can enlarge the receptive field of a network and improve its adaptability. It achieves excellent results in image classification and certain downstream tasks such as instance segmentation, object detection and keypoint detection.

### *6.11 SPNet*

Spatial pooling usually operates on a small region which limits its capability to capture long-range dependencies and focus on distant regions. To overcome this, Hou proposed *strip pooling*, a novel pooling method capable of encoding long-range context in either horizontal or vertical spatial domains.

Strip pooling has two branches for horizontal and vertical strip pooling. The strip pooling module (SPM) is further developed in the mixed pooling module (MPM). Both consider spatial and channel relationships to overcome the locality of convolutional neural networks. SPNet achieves state-of-the-art results for several complex semantic segmentation benchmarks.

### *6.12 SCA-CNN*

As CNN features are naturally spatial, channel-wise and multi-layer, Chen proposed a novel *spatial and channel-wise attention-based convolutional neural network* (SCACNN). It was designed for the task of image captioning, and uses an encoder-decoder framework where a CNN first encodes an input image into a vector and then an LSTM decodes the vector into a sequence of words. Given an input feature map $X$ and the previous time step LSTM hidden state $h_{t-1} \in R^d$, a spatial attention mechanism pays more attention to the semantically useful regions, guided by LSTM hidden state $h_{t-1}$. The spatial attention model is:

$$a(h_{t-1},X) = \tanh(\text{Conv}_1^{1\times1}(X) \oplus W_1 h_{t-1}) \quad (136)$$

$$\Phi_s(h_{t-1},X) = \text{Softmax}(\text{Conv}_2^{1\times1}(a(h_{t-1},X))) \quad (137)$$

where $\oplus$ represents addition of a matrix and a vector. Similarly, channel-wise attention aggregates global information first, and then computes a channel-wise attention weight vector with the hidden state $h_{t-1}$:

$$b(h_{t-1},X) = \tanh((W_2\text{GAP}(X) + b_2) \oplus W_1 h_{t-1}) \quad (138)$$

$$\Phi_c(h_{t-1},X) = \text{Softmax}(W_3(b(h_{t-1},X)) + b_3) \quad (139)$$

Unlike previous attention mechanisms which consider each image region equally and use global spatial information to tell the network where to focus, SCA-Net leverages the semantic vector to produce the spatial attention map as well as the channel-wise attention weight vector. Being more than a powerful attention model, SCA-CNN also provides a better understanding of where and what the model should focus on during sentence generation.

### *6.13 GALA*

Most attention mechanisms learn where to focus using only weak supervisory signals from class labels, which inspired Linsley to investigate how explicit human supervision can affect the performance and interpretability of attention models. As a proof of concept, Linsley proposed the *global-and-local attention* (GALA) module, which extends an SE block with a spatial attention mechanism.

Given the input feature map $X$, GALA uses an attention mask that combines global and local attention to tell the network where and on what to focus. As in SE blocks, global attention aggregates global information by global average pooling and then produces a channel-wise attention weight vector using a multilayer perceptron. In local attention, two consecutive $1 \times 1$ convolutions are conducted on the input to produce a positional weight map. The outputs of the local and global pathways are combined by addition and multiplication.

Supervised by human-provided feature importance maps, GALA has significantly improved representational power and can be combined with any CNN backbone.

Representative channel & spatial attention mechanisms sorted by date. Cls = classification, ICap = image captioning, Det = detection, Seg = segmentation, ISeg = instance segmentation, KP = keypoint detection, ReID = re-identification.$g(x)$ and $f(g(x),x)$ are the attention process described by Eq. 1. Ranges means the ranges of attention map. S or H means soft or hard attention. (A) element-wise product. (B) aggregate information via attention map.(I) focus the network on the discriminative region, (II) emphasize important channels, (III) capture long-range information, (IV) capture cross-domain interaction between any two domains.

| Category | Method | Publication | Tasks | $g(x)$ | $f(g(x),x)$ | Ranges | SorH | Goals |
|---|---|---|---|---|---|---|---|---|
| Jointly predict channel & spatial attention map | Residual Attention [119] | CVPR2017 | Cls | top-down network -> bottom down network -> 1×1 Convs -> Sigmoid | (A) | (0,1) | S | (I), (II) |
| | SCNet [120] | CVPR2020 | Cls, Det, ISeg, KP | top-down network -> bottom down network -> identity add -> sigmoid | (A) | (0,1) | S | (II), (III) |
| | Strip Pooling [124] | CVPR2020 | Seg | a)horizontal/vertical global pooling -> 1D Conv -> point-wise summation -> $1 \times 1$ Conv -> Sigmoid | (A) | (0,1) | S | (I), (II), (III) |
| Separately predict channel & spatial attention maps | SCA-CNN [50] | CVPR2017 | ICap | a)spatial: fuse hidden state -> $1 \times 1$ Conv -> Softmax, b)channel: global average pooling -> MLP -> Softmax | (A) | (0,1) | S | (I), (II), (III) |
| | CBAM [6] | ECCV2018 | Cls, Det | a)spatial: global pooling in channel dimension-> Conv -> Sigmoid, b)channel: global pooling in spatial dimension -> MLP -> Sigmoid | (A) | (0,1) | S | (I), (II), (III) |
| | BAM [6] | BMVC2018 | Cls, Det | a)spatial: dilated Convs, b)channel: global average pooling -> MLP, c)fuse two branches | (A) | (0,1) | S | (I), (II), (III) |
| | scSE [123] | TMI2018 | Seg | a)spatial: $1 \times 1$ Conv -> Sigmoid, b)channel: global average pooling -> MLP -> Sigmoid, c)fuse two branches | (A) | (0,1) | S | (I), (II), (III) |
| | Dual Attention [10] | CVPR2019 | Seg | a)spatial: self-attention in spatial dimension, b)channel: self-attention in channel dimension, c) fuse two branches | (B) | (0,1) | S | (I), (II), (III) |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| RGA [101] | CVPR2020 | ReID | use self-attention to capture pairwise relations -> compute attention maps with the input and relation vectors | (A) | (0,1) | S | (I), (II), (III) |
| Triplet Attention [121] | WACV2021 | Cls, Det | compute attention maps for pairs of domains -> fuse different branches | (A) | (0,1) | S | (I), (IV) |

## 7 Spatial & Temporal Attention

Spatial & temporal attention combines the advantages of spatial attention and temporal attention as it adaptively selects both important regions and key frames. Some works compute temporal attention and spatial attention separately, while others produce joint spatiotemporal attention maps. Further works focusing on capturing pairwise relation . Representative spatial & temporal attention attentions and specific process

### *7.1 STA-LSTM*

In human action recognition, each type of action generally only depends on a few specific kinematic joints . Furthermore, over time, multiple actions may be performed. Motivated by these observations, Song proposed a joint spatial and temporal attention network based on LSTM , to adaptively find discriminative features and keyframes. Its main attention-related components are a spatial attention sub-network, to select important regions, and a temporal attention sub-network, to select key frames.

It adopts a ReLU function instead of a normalization function for ease of optimization. It also uses a regularized objective function to improve convergence.

Overall, this presents a joint spatiotemporal attention method to focus on important joints and keyframes, with excellent results on the action recognition task.

### *7.2 RSTAN*

To capture spatiotemporal contexts in video frames, Du introduced *spatiotemporal attention* to adaptively identify key features in a global way.

The spatiotemporal attention mechanism in RSTAN consists of a spatial attention module and a temporal attention module applied serially. Given an input feature map $X \in \mathrm{R}^{D\times T\times H\times W}$ and the previous hidden state $h_{t-1}$ of an RNN model, spatiotemporal attention aims to produce a spatiotemporal feature representation for action recognition. First, the given feature map $X$ is reshaped to $\mathrm{R}^{D\times T\times (H\times W)}$, and we define $X(n,k)$ as the feature vector for the $k$-th location of the $n$-th frame. At time $t$, the spatial attention mechanism aims to produce a global feature $l_n$ for each frame

The spatiotemporal attention mechanism used in RSTAN identifies those regions in both spatial and temporal domains which are strongly related to the prediction in the current step of the RNN. This efficiently enhances the representation power of any 2D CNN.

### *7.3 STA*

Previous attention-based methods for video-based person re-identification only assigned an attention weight to each frame and failed to capture joint spatial and temporal relationships. To address this issue, Fu propose a novel *spatiotemporal attention* (STA) approach, which assigns attention scores for each spatial region in different frames without any extra parameters.

Instead of computing spatial attention maps frame by frame, STA considers spatial and temporal attention information simultaneously, fully using the discriminative parts in both dimensions. This reduces the influence of occlusion. Because of its non-parametric design, STA can tackle input video sequences of variable length; it can be combined with any 2D CNN backbone.

### *7.4 STGCN*

To model the spatial relations within a frame and temporal relations across frames, Yang proposed a novel *spatiotemporal graph convolutional network* (STGCN) to learn a discriminative descriptor for a video. It constructs a patch graph using pairwise similarity, and then uses graph convolution to aggregate information.

STGCN includes two parallel GCN branches, the temporal graph module and the structural graph module. Given the feature maps of a video, STGCN first horizontally partitions each frame into $P$ patches and applies average pooling to generate patch-wise features $x_1,\ldots,x_N$, where the total number of patches is $N = TP$. For the temporal module, it takes each patch as a graph node and construct a patch graph for the video, where the adjacency matrix $A_b$ is obtained by normalizing the pairwise relation matrix $E$,

For the spatial module, STGCN follows a similar approach of adjacency matrix and graph convolution, except for modeling the spatial relations of different regions within a frame.

Flattening spatial and temporal dimensions into a sequence, STGCN applies the GCN to capture the spatiotemporal relationships of patches across different frames. Pairwise attention is used to obtain the weighted adjacency matrix. By leveraging spatial and temporal relationships between patches, STGCN overcomes the occlusion problem while also enhancing informative features. It can used with any CNN backbone to process video.

TABLE 8

Representative spatial & temporal attentions sorted by date. Action=action recognition, ReID = re-identification. Ranges means the ranges of attention map. S or H means soft or hard attention. $g(x)$ and $f(g(x),x)$ are the attention process described by Eq. 1. (A)element-wise product.(B) aggregate information via attention map.(I) emphasize key points in both spatial and temporal domains, (II)capture global information.

| Category | Method | Publication | Tasks | $g(x)$ | $f(g(x),x)$ | Ranges | SorH | Goals |
|---|---|---|---|---|---|---|---|---|
| Separately predict spatial & temporal attention | STA-LSTM [130] | AAAI2017 | Action | a)spatial: fuse hidden state -> MLP -> Softmax, b)temporal: fuse hidden state -> MLP -> ReLU | (A) | (0,1), (0,+∞) | S | (I) |
| | RSTAN [16] | TIP2018 | Action | a)spatial: fuse hidden state -> MLP -> Softmax, b)temporal: fuse hidden state -> MLP -> Softmax | (B) | (0,1) | S | (I) (II) |
| Jointly predict spatial & temporal attention | STA [131] | AAAI2019 | ReID | a) tenporal: produce perframe attention maps using $l_2$ norm b) spatial: obtain spatial scores for each patch by summation using $l_1$ norm. | (B) | (0,1) | S | (I) |
| Pairwise relation-based method | STGCN [177] | CVPR2020 | ReID | construct a patch graph using pairwise similar-ity | (B) | (0,1) | S | (I) |