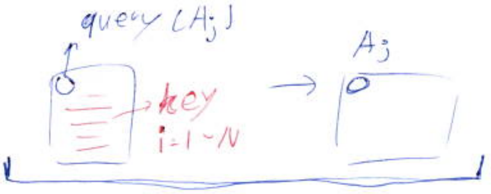

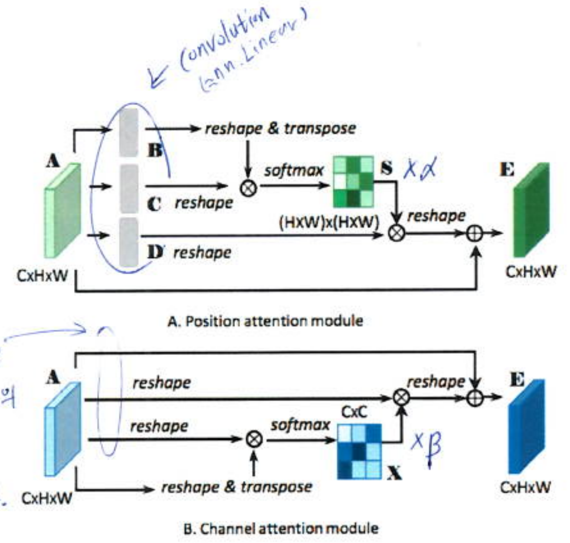

conv 안 함!
why? channel 사이의 관계를 파악하는 데 방해될까봐.

Figure 3: The details of Position Attention Module and Channel Attention Module are illustrated in (A) and (B). (Best viewed in color)

(세개씩) similarity attention score 를 왜 channel 끼리 해도 좋은지에 대한 얘기X 無

the first step, in which channel attention matrix is calculated in channel dimension. Finally we aggregate the outputs from the two attention modules to obtain better feature representations for pixel-level prediction.

3.2. Position Attention Module

Discriminant feature representations are essential for scene understanding, which could be obtained by capturing long-range contextual information. However, many works [15, 29] suggest that local features generated by traditional FCNs could lead to misclassification of objects and stuff. In order to model rich contextual relationships over local features, we introduce a position attention module. The position attention module encodes a wider range of contextual information into local features, thus enhancing their representation capability. Next, we elaborate the process to adaptively aggregate spatial contexts.

As illustrated in Figure.3(A), given a local feature $\mathbf{A} \in \mathbb{R}^{C \times H \times W}$, we first feed it into a convolution layers to generate two new feature maps $\mathbf{B}$ and $\mathbf{C}$, respectively, where $\{\mathbf{B}, \mathbf{C}\} \in \mathbb{R}^{C \times H \times W}$. Then we reshape them to $\mathbb{R}^{C \times N}$, where $N = H \times W$ is the number of pixels. After that we perform a matrix multiplication between the transpose of $\mathbf{C}$ and $\mathbf{B}$, and apply a softmax layer to calculate the spatial attention map $\mathbf{S} \in \mathbb{R}^{N \times N}$:

$$s_{ji} = \frac{exp(B_i \cdot C_j)}{\sum_{i=1}^{N} exp(B_i \cdot C_j)} \qquad (1)$$

where s_{ji} measures the i^{th} position's impact on j^{th} position. The more similar feature representations of the two position contributes to greater correlation between them.

Meanwhile, we feed feature $\mathbf{A}$ into a convolution layer to generate a new feature map $\mathbf{D} \in \mathbb{R}^{C \times H \times W}$ and reshape it to $\mathbb{R}^{C \times N}$. Then we perform a matrix multiplication between $\mathbf{D}$ and the transpose of $\mathbf{S}$ and reshape the result to $\mathbb{R}^{C \times H \times W}$. Finally, we multiply it by a scale parameter α and perform a element-wise sum operation with the features $\mathbf{A}$ to obtain the final output $\mathbf{E} \in \mathbb{R}^{C \times H \times W}$ as follows:

$$E_j = \alpha \sum_{i=1}^{N} (s_{ji} D_i) + A_j \leftarrow \text{query} \qquad (2)$$

where α is initialized as 0 and gradually learns to assign more weight [28]. It can be inferred from Equation 2 that the resulting feature $\mathbf{E}$ at each position is a weighted sum of the features across all positions and original features. Therefore, it has a global contextual view and selectively aggregates contexts according to the spatial attention map. The similar semantic features achieve mutual gains, thus imporving intra-class compact and semantic consistency.

3.3. Channel Attention Module

Each channel map of high level features can be regarded as a class-specific response, and different semantic responses are associated with each other. By exploiting the interdependencies between channel maps, we could emphasize interdependent feature maps and improve the feature representation of specific semantics. Therefore, we build a channel attention module to explicitly model interdependencies between channels.

상호의존성

1. 한 channel 즉 특정 class-specific channel 이 query 가 되고
2. (keys) 다른 class channel 에게 질문!
3. 종합 결과 도출 된a

The structure of channel attention module is illustrated in Figure.3(B). Different from the position attention module, we directly calculate the channel attention map $\mathbf{X} \in \mathbb{R}^{C \times C}$ from the original features $\mathbf{A} \in \mathbb{R}^{C \times H \times W}$. Specifically, we reshape $\mathbf{A}$ to $\mathbb{R}^{C \times N}$, and then perform a matrix multiplication between $\mathbf{A}$ and the transpose of $\mathbf{A}$. Finally, we apply a softmax layer to obtain the channel attention map $\mathbf{X} \in \mathbb{R}^{C \times C}$:

$$x_{ji} = \frac{exp(A_i \cdot A_j)}{\sum_{i=1}^{C} exp(A_i \cdot A_j)} \qquad (3)$$

where x_{ji} measures the i^{th} channel's impact on the j^{th} channel. In addition, we perform a matrix multiplication between the transpose of $\mathbf{X}$ and $\mathbf{A}$ and reshape their result to $\mathbb{R}^{C \times H \times W}$. Then we multiply the result by a scale parameter β and perform an element-wise sum operation with $\mathbf{A}$ to obtain the final output $\mathbf{E} \in \mathbb{R}^{C \times H \times W}$:

$$E_j = \beta \sum_{i=1}^{C} (x_{ji} A_i) + A_j \qquad (4)$$

where β gradually learns a weight from 0. The Equation 4 shows that the final feature of each channel is a weighted sum of the features of all channels and original features, which models the long-range semantic dependencies between feature maps. It helps to boost feature discriminability.

이 의미들에 따라 생기는 feature 의 차별성, 다름 강조