

Attention Total Variation Loss

ATV Loss in StableVITON

- Objective of ATV loss in StableVITON
 - Cross-attention layer 는 옷을 agnostic map 에 할당해야한다.
 - (b): 색깔 불일치가 발생



Figure 4. Visualization of attention map from a zero cross-attention block of 32 resolution.

ATV Loss in StableVITON

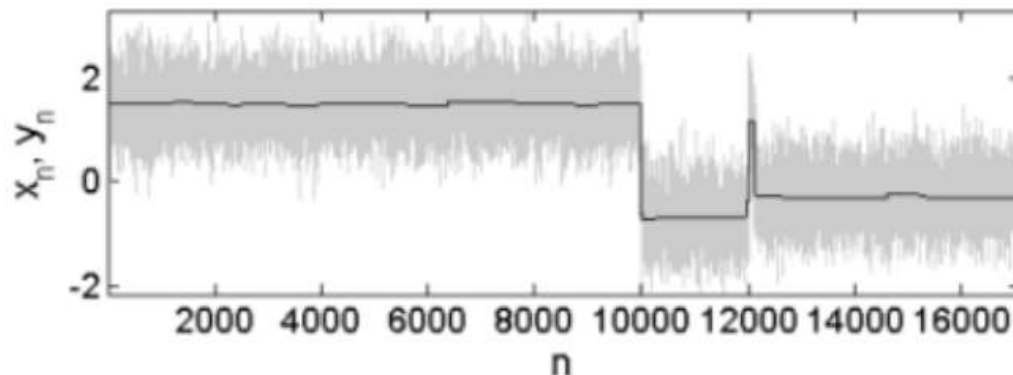
- Objective of ATV loss in StableVITON
 - 균일하게 분포된 attention map 에 중심 좌표를 적용하여, 분산되어 위치한 attention score 간의 간섭을 완화하도록 설계됐다.
 - 즉, attention score 간의 smoothing?

where $M \in \{0, 1\}^{H_q \times W_q}$ is the ground truth clothing mask to only affect the clothing region. The attention total variation loss \mathcal{L}_{ATV} is designed to enforce the center coordinates on the attention map uniformly distributed, thereby alleviating interference among attention scores located at dispersed positions. As illustrated in Figure (c), this leads to the gen-

ATV Loss in StableVITON

- Total variation denoising

1D Signal



Noise한 입력 신호 x_n (=회색선)에 디지털 신호 y_n (=검정선)를 근사(=E)시키고, y_n 과 y_{n+1} 에 대한 차이를 작게 (=V)하는 것을 목표로 합니다. x_n 과 y_n 의 관계식은 다음과 같이 표현됩니다.

$$E(x, y) = \frac{1}{n} \sum_n (x_n - y_n)^2 \quad V(y) = \sum_n |y_{n+1} - y_n|$$

ATV Loss in StableVITON

- ATV loss

center coordinate map $F \in \mathbb{R}^{H_q \times W_q \times 2}$ as follows:

$$F_{ijn} = \frac{1}{h_k w_k} \sum_{k=1}^{h_k} \sum_{l=1}^{w_k} (A_{ijkl} \odot G_{kln}), \quad (4)$$

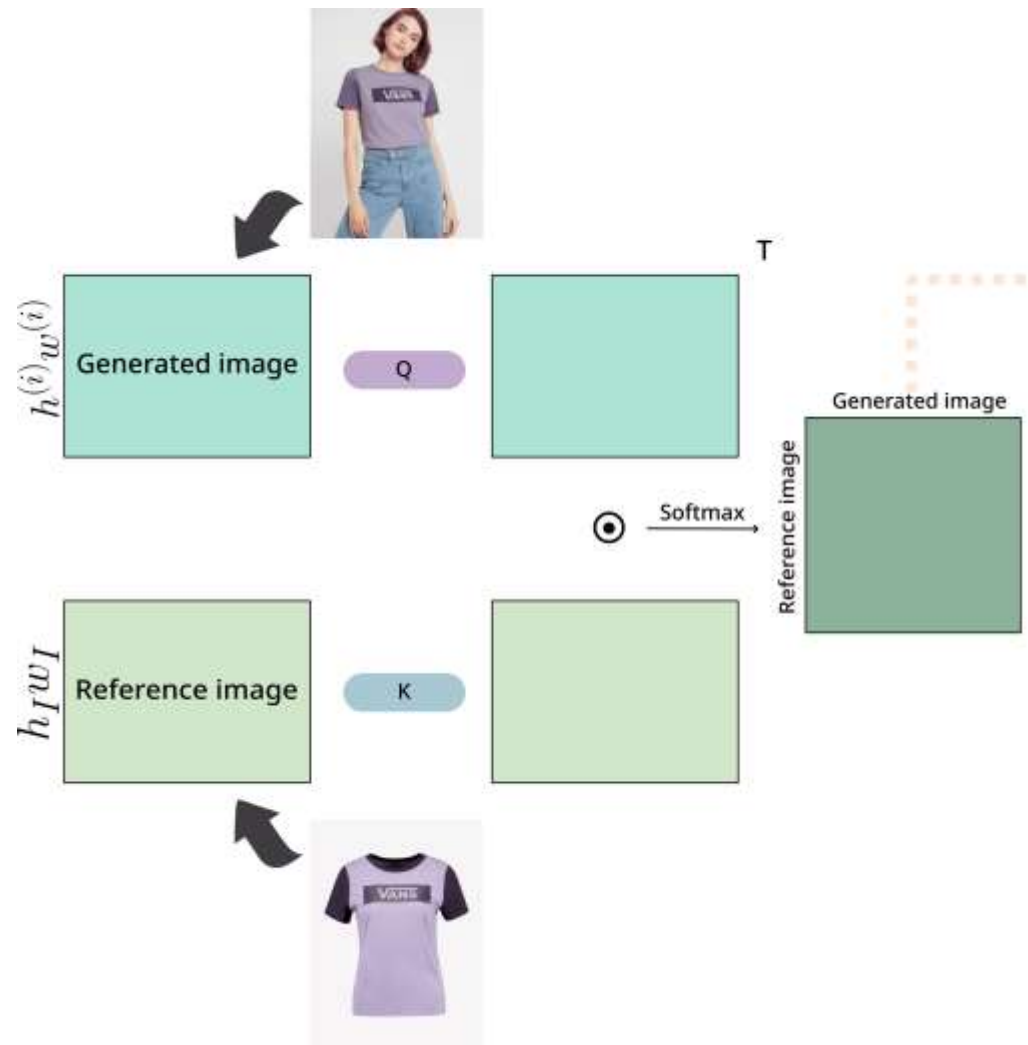
where we average the attention map over the head dimension and reshape it as $A \in \mathbb{R}^{H_q \times W_q \times h_k \times w_k}$, and $G \in [-1, 1]^{h_k \times w_k \times 2}$ is a 2D normalized coordinate. \odot indicates element-wise multiplication operation.

For each query token in each clothing-agnostic region, the center coordinates should be evenly distributed, and the attention total variation loss \mathcal{L}_{ATV} is defined as follows:

$$\mathcal{L}_{ATV} = \| \nabla(F \odot M) \|_1, \quad (5)$$

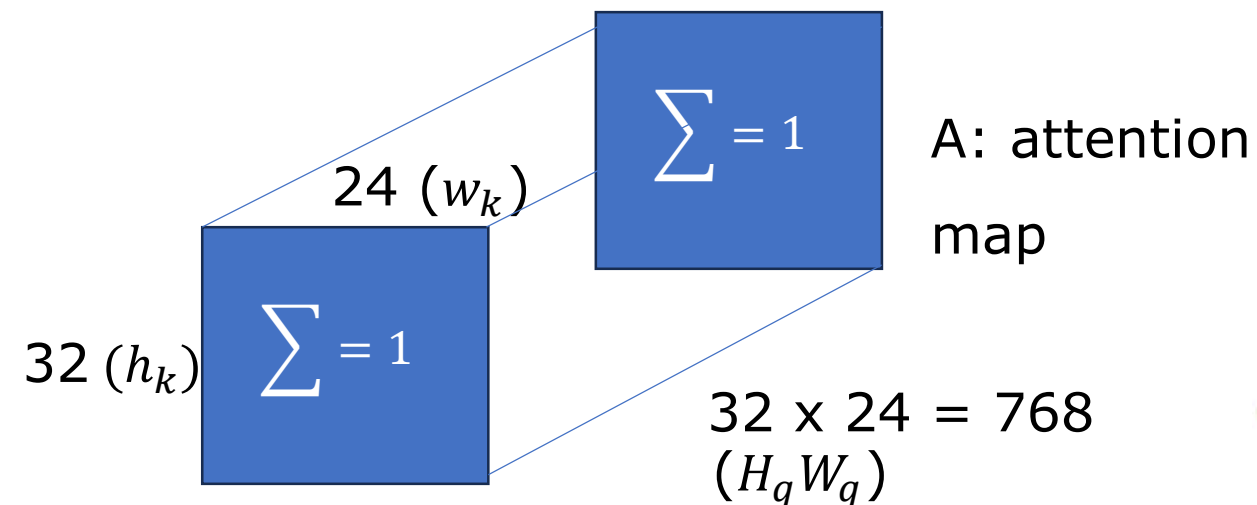
ATV Loss in StableVITON

- Attention map
 - $(h^{(i)}w^{(i)}) \times (h_Iw_I)$ 을 아래와 같이 생각
 - $HW \times hw$
 - 즉, HW는 generated, hw는 reference



ATV Loss in StableVITON

- ATV loss: Center coordinate map



Generated patch i 에 영향을 끼친
reference image 의 attention map
이 768 개 있다.

center coordinate map $F \in \mathbb{R}^{H_q \times W_q \times 2}$ as follows:

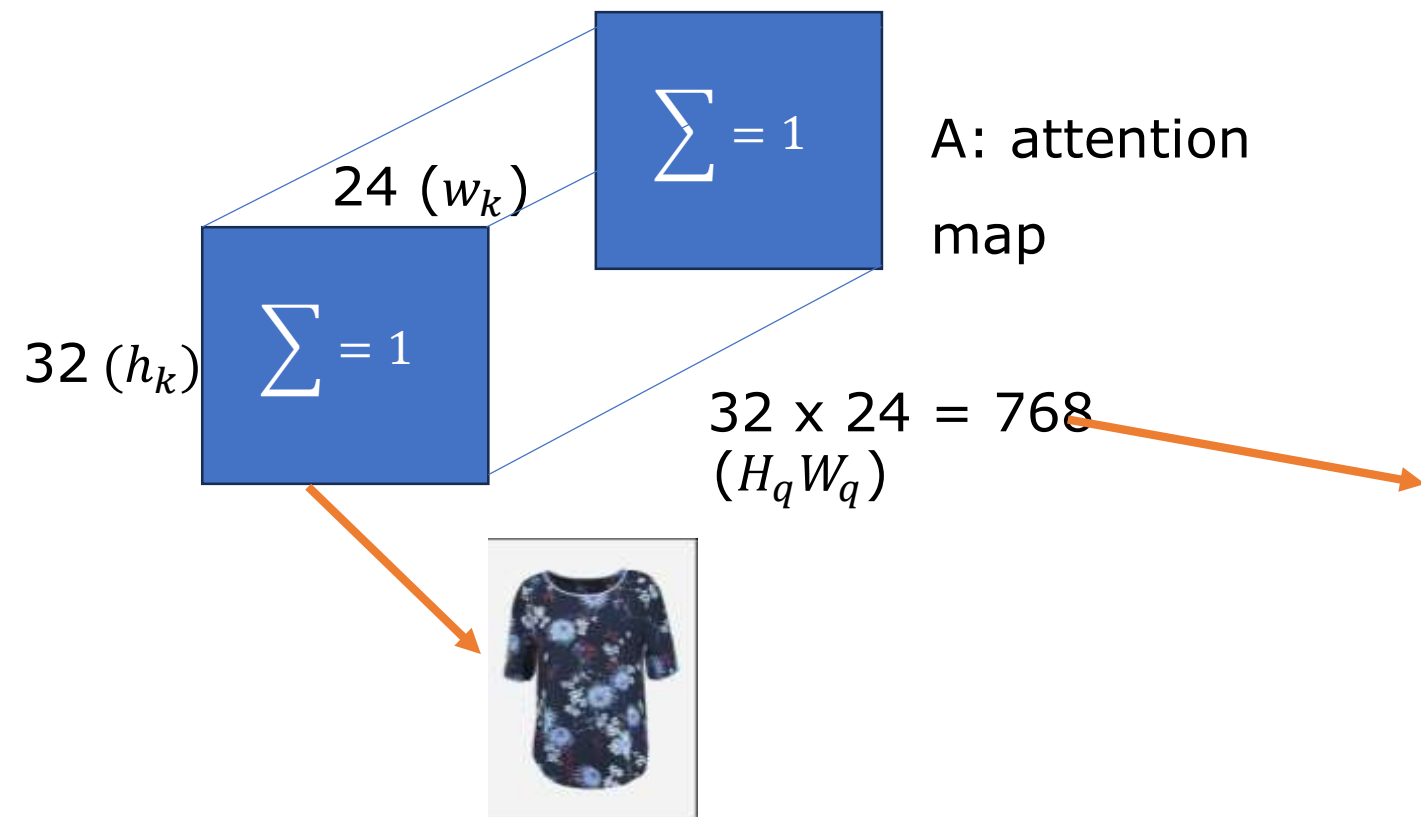
$$F_{ijn} = \frac{1}{h_k w_k} \sum_{k=1}^{h_k} \sum_{l=1}^{w_k} (A_{ijkl} \odot G_{kl n}), \quad (4)$$

where we average the attention map over the head dimension and reshape it as $A \in \mathbb{R}^{H_q \times W_q \times h_k \times w_k}$, and $G \in [-1, 1]^{h_k \times w_k \times 2}$ is a 2D normalized coordinate. \odot indicates element-wise multiplication operation.

ATV Loss in StableVITON

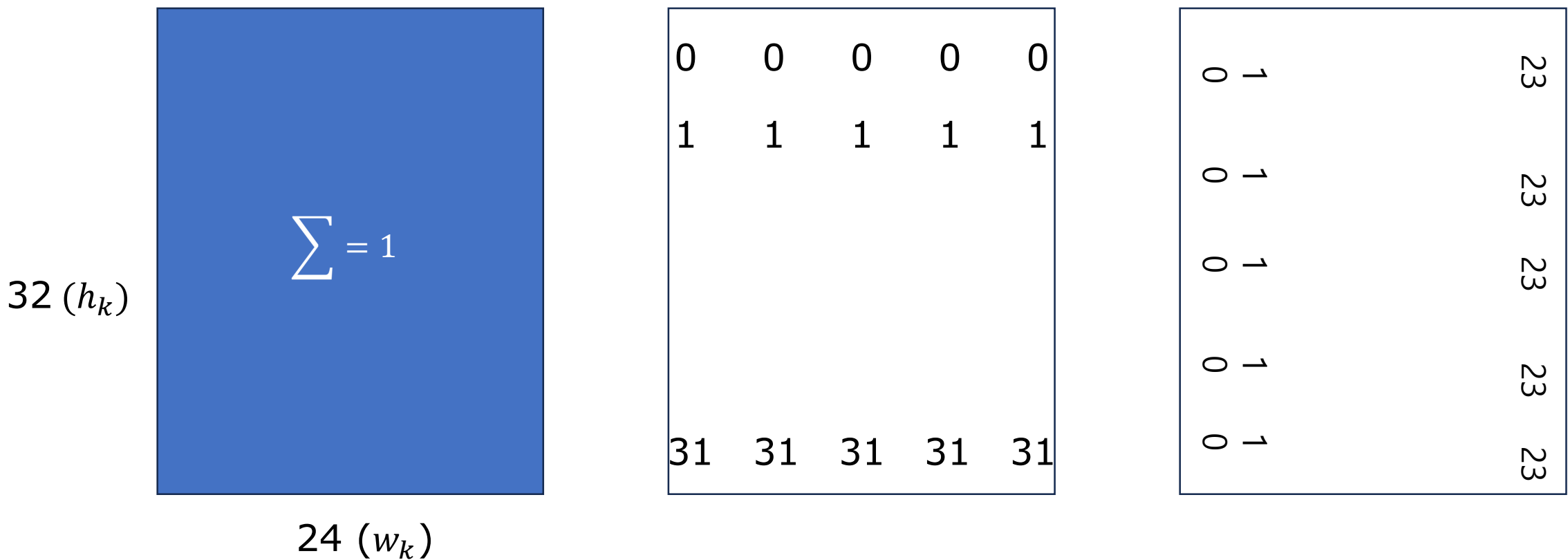
- ATV loss: Center coordinate map

Generated patch i 에 영향을 끼친
reference image 의 attention map
이 768 개 있다.



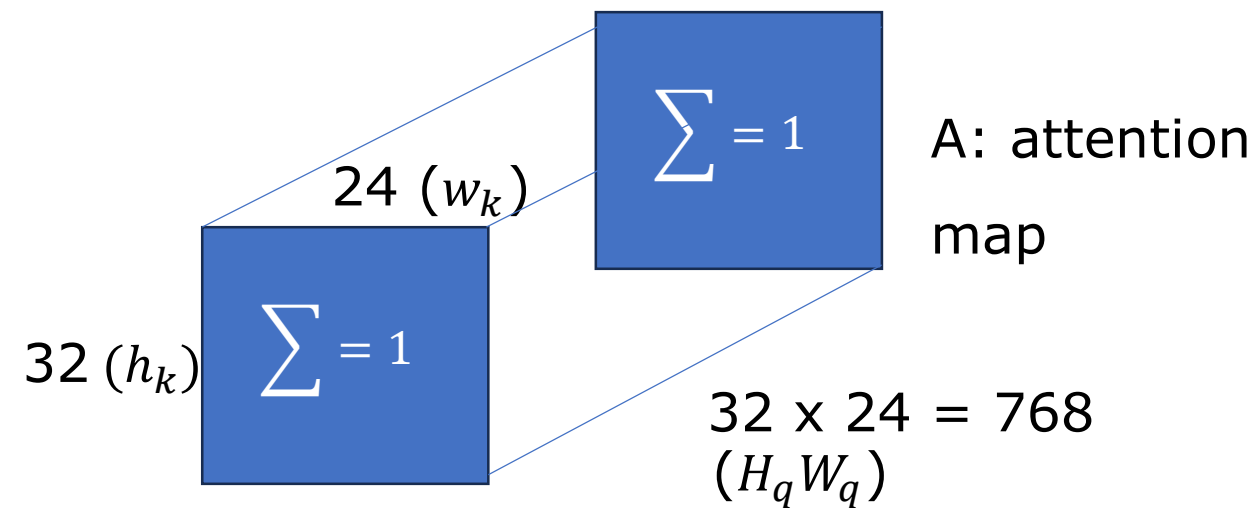
ATV Loss in StableVITON

- ATV loss: Center coordinate map



ATV Loss in StableVITON

- ATV loss: Center coordinate map

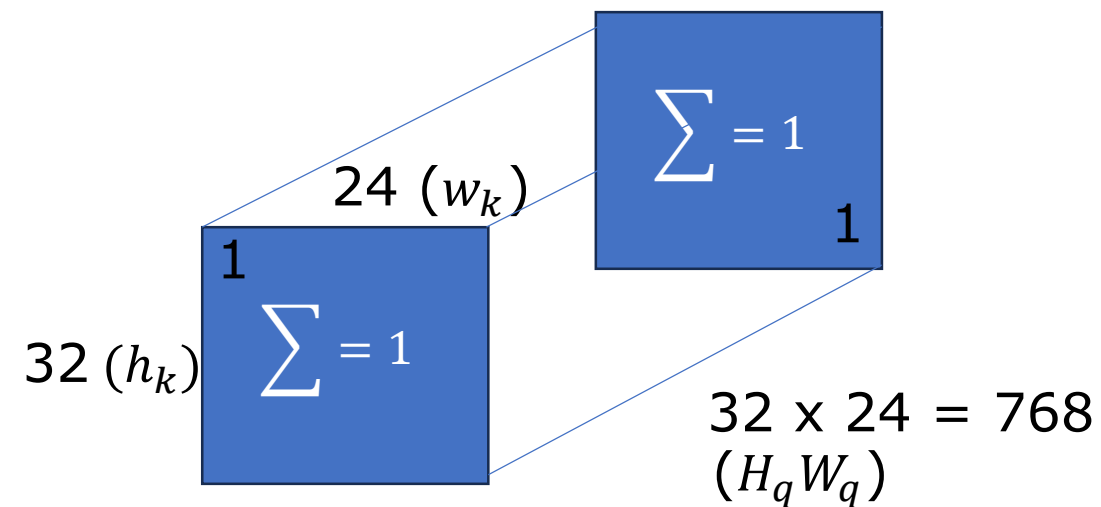


0	0	0	0	0
1	1	1	1	1
31	31	31	31	31

0	1				23
0	1				23
0	1				23
0	1				23
0	1				23

ATV Loss in StableVITON

- ATV loss: Center coordinate map



$$\begin{matrix} 1 \\ \sum = 1 \end{matrix}$$

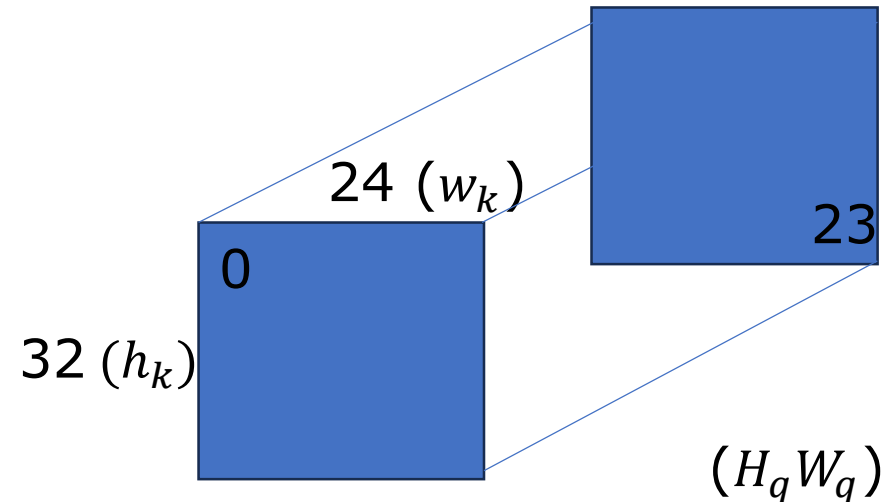
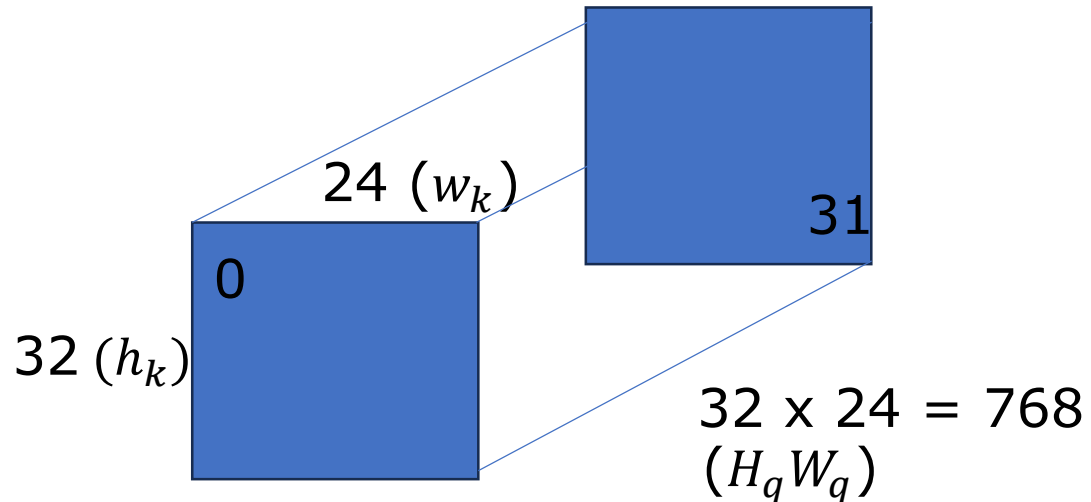
맨 앞 channel의 $(0,0)$ 위치에서의 attention value를 1이라고 가정
즉, 나머지 원소의 값은 0이다.

$$\begin{matrix} \sum = 1 \\ 1 \end{matrix}$$

맨 뒤 channel의 (h,w) 위치에서의 attention value를 1이라고 가정

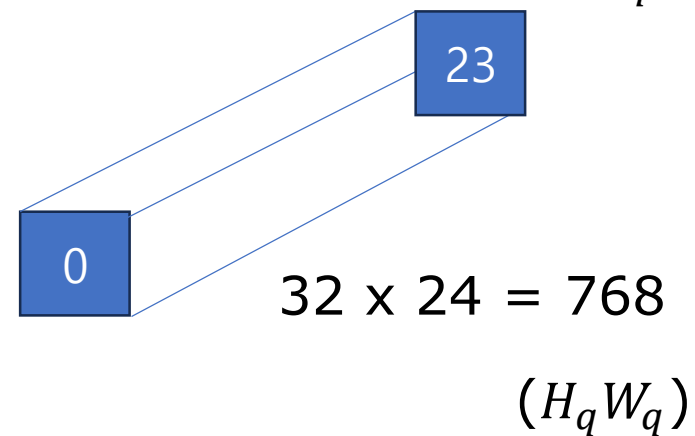
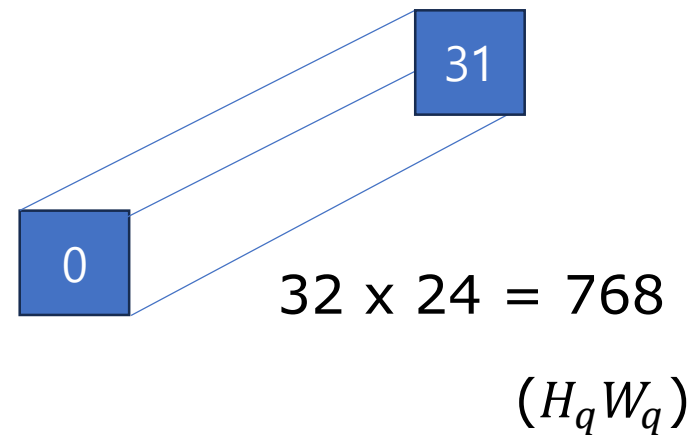
ATV Loss in StableVITON

- ATV loss: Center coordinate map
 - 768 channel의 하나하나가
 - 세로 위치가 곱해진 reference attention map / 가로 위치가 곱해진 reference attention map
 - $BS \times H_q W_q \times h_k \times w_k \times 2$



ATV Loss in StableVITON

- ATV loss: Center coordinate map
 - $h_k \times w_k$ 에 대해서 평균을 낸다.
 - $BS \times H_q W_q \times 2$

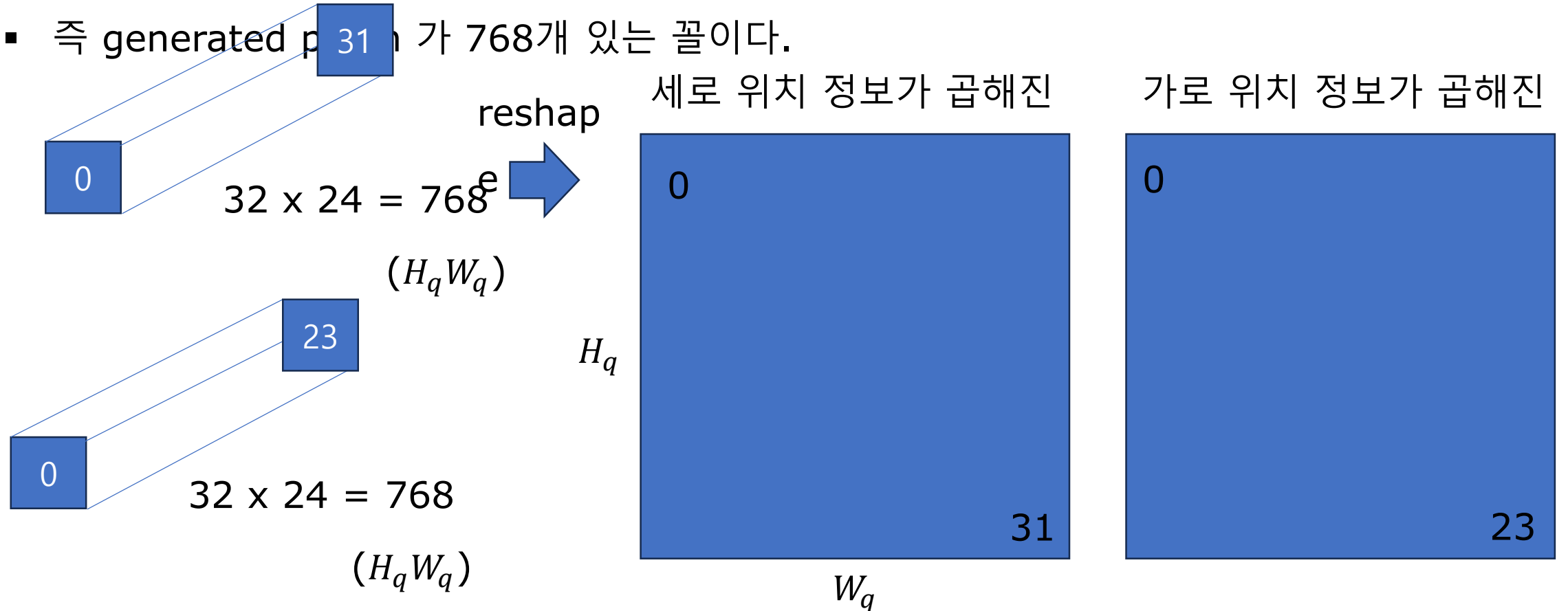


ATV Loss in StableVITON

- ATV loss: Center coordinate map

- Generated patch i 에 영향을 주는 reference image 의 평균적인 attention score가 768개

- 즉 generated patch i 가 768개 있는 꼴이다.



ATV Loss in StableVITON

- ATV loss: Center coordinate map
 - Generated patch i 에 영향을 주는 위치 정보가 곱해진 reference image 의 평균적인 attention score
 - 실제 의미가 있는 (warped clothing region) 에 대해서만 variation 을 줄인다

center coordinate map $F \in \mathbb{R}^{H_q \times W_q \times 2}$ as follows:

$$F_{ijn} = \frac{1}{h_k w_k} \sum_{k=1}^{h_k} \sum_{l=1}^{w_k} (A_{ijkl} \odot G_{kl n}), \quad (4)$$

where we average the attention map over the head dimension and reshape it as $A \in \mathbb{R}^{H_q \times W_q \times h_k \times w_k}$, and $G \in [-1, 1]^{h_k \times w_k \times 2}$ is a 2D normalized coordinate. \odot indicates element-wise multiplication operation.

For each query token in each clothing-agnostic region, the center coordinates should be evenly distributed, and the attention total variation loss \mathcal{L}_{ATV} is defined as follows:

$$\mathcal{L}_{ATV} = \| \nabla(F \odot M) \|_1, \quad (5)$$

0

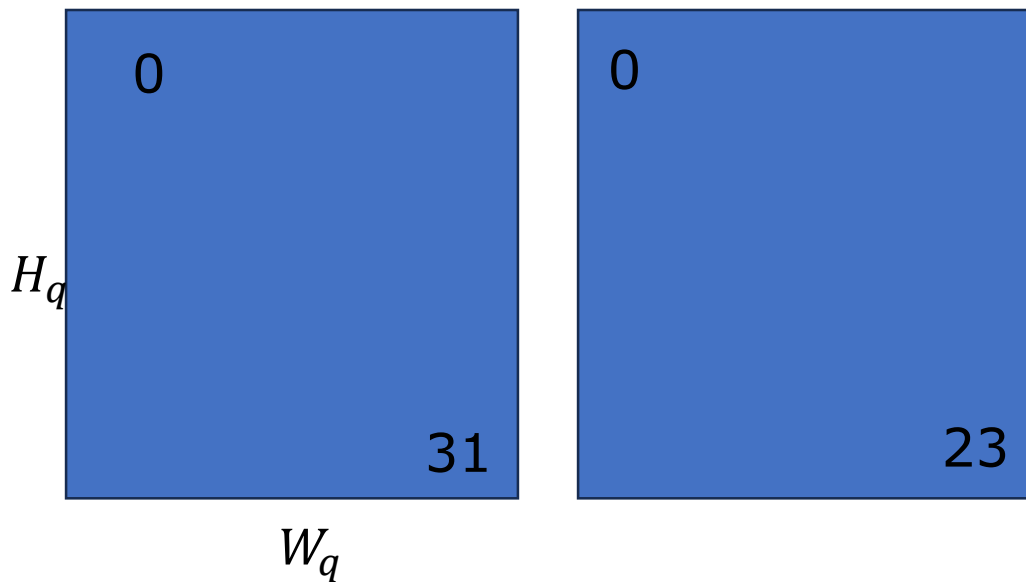
0

31

23

ATV Loss in StableVITON

- ATV loss: Center coordinate map
 - 실제 의미가 있는 (warped clothing region) 에 대해서만 variation 을 줄인

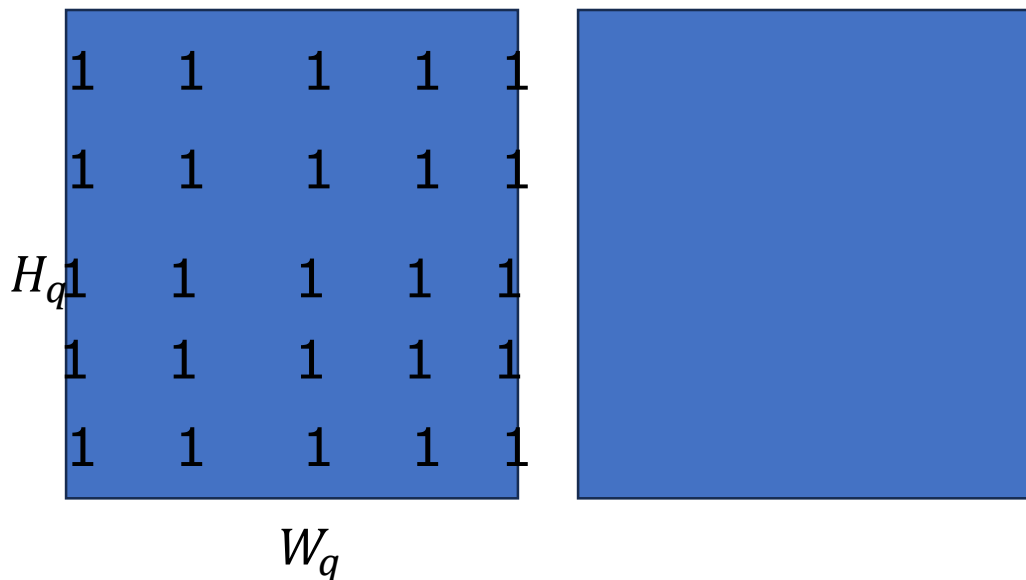


```
def get_tvloss(coords, mask, ch, cw):  
    b, n, _ = coords.shape  
    ← coords = coords.reshape(b, ch, cw, 2)  
    mask = mask.unsqueeze(-1)  
    y_mask = mask[:, 1:] * mask[:, :-1]  
    x_mask = mask[:, :, 1:] * mask[:, :, :-1]  
  
    y_tvloss = torch.abs(coords[:, 1:] - coords[:, :-1]) * y_mask  
    x_tvloss = torch.abs(coords[:, :, 1:] - coords[:, :, :-1]) * x_mask  
    tv_loss = y_tvloss.sum() / y_mask.sum() + x_tvloss.sum() / x_mask.sum()  
    return tv_loss
```

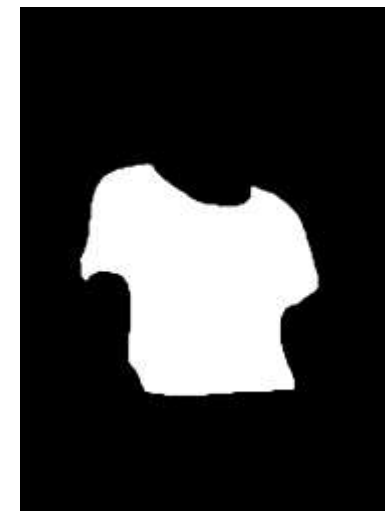
Diagram illustrating the mask used in the ATV loss calculation. The mask is a white silhouette of a t-shirt on a black background. An arrow points from the 'mask' variable in the code to this image.

ATV Loss in StableVITON

- ATV loss: Center coordinate map
 - 위치 정보가 곱해져있지 않았다면 모든 원소들은 1 이었으므로 위치 정보가 곱해진 이유가 존재한다.

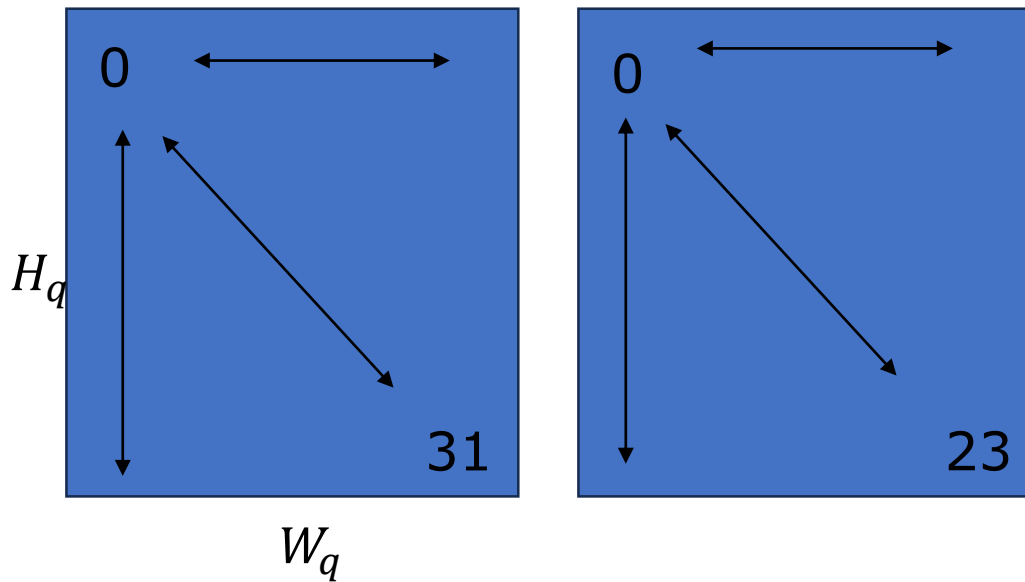


```
def get_tvloss(coords, mask, ch, cw):  
    b, n, _ = coords.shape  
    ← coords = coords.reshape(b, ch, cw, 2)  
    mask = mask.unsqueeze(-1)  
    y_mask = mask[:, 1:] * mask[:, :-1]  
    x_mask = mask[:, :, 1:] * mask[:, :, :-1]  
  
    y_tvloss = torch.abs(coords[:, 1:] - coords[:, :-1]) * y_mask  
    x_tvloss = torch.abs(coords[:, :, 1:] - coords[:, :, :-1]) * x_mask  
    tv_loss = y_tvloss.sum() / y_mask.sum() + x_tvloss.sum() / x_mask.sum()  
    return tv_loss
```

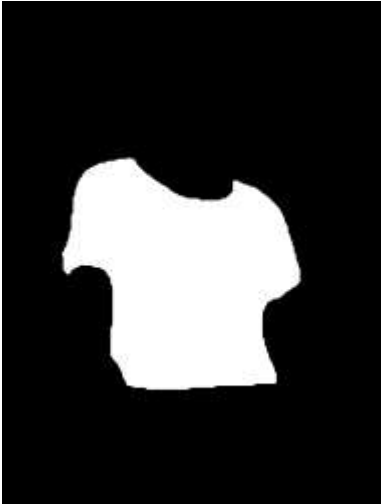


ATV Loss in StableVITON

- 직관적인 해석?
 - 우리가 집중해야 하는 generated patch i 들에 대해,
위치 정보를 곱한 reference image 의 attention score 의 가중 평균 값들의 변화량이 적어야 한다.

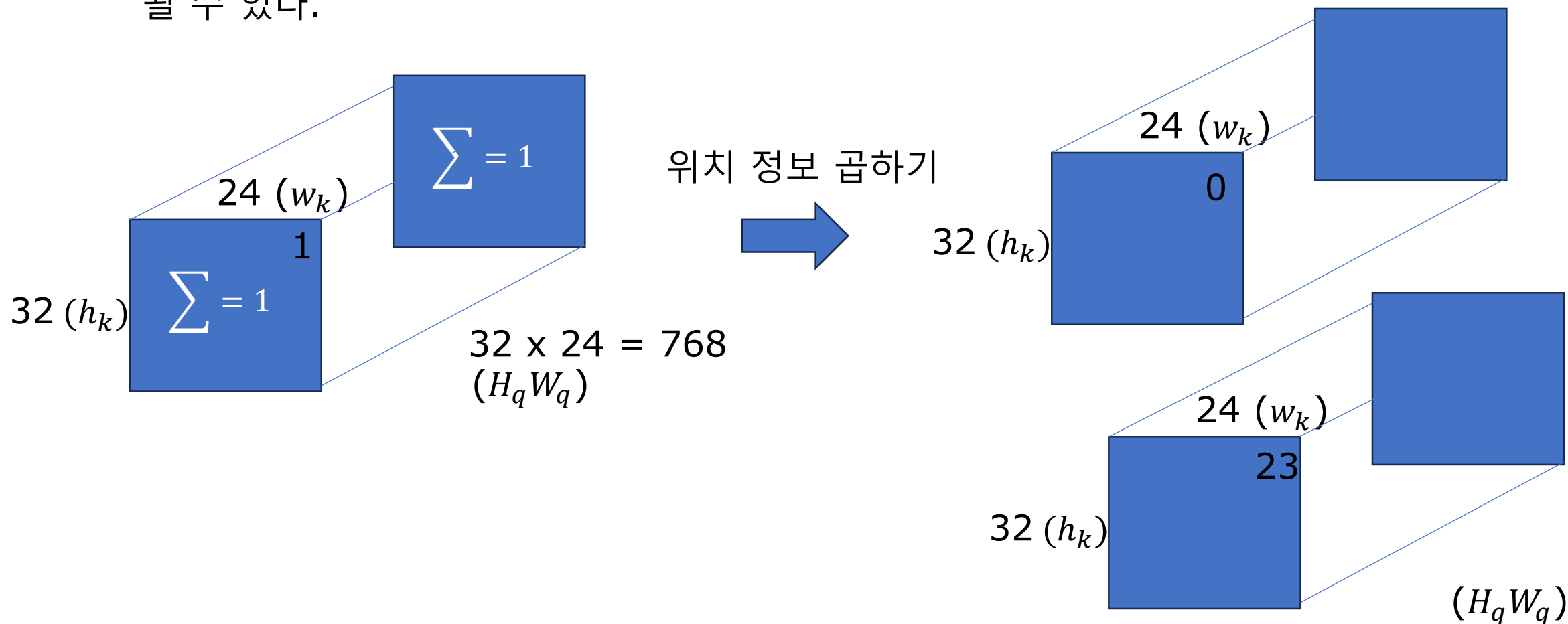


```
def get_tvloss(coords, mask, ch, cw):  
    b, n, _ = coords.shape  
    ← coords = coords.reshape(b, ch, cw, 2)  
    mask = mask.unsqueeze(-1)  
    y_mask = mask[:, 1:] * mask[:, :, -1]  
    x_mask = mask[:, :, 1:] * mask[:, :, -1]  
  
    y_tvloss = torch.abs(coords[:, 1:] - coords[:, :, -1]) * y_mask  
    x_tvloss = torch.abs(coords[:, :, 1:] - coords[:, :, -1]) * x_mask  
    tv_loss = y_tvloss.sum() / y_mask.sum() + x_tvloss.sum() / x_mask.sum()  
    return tv_loss
```



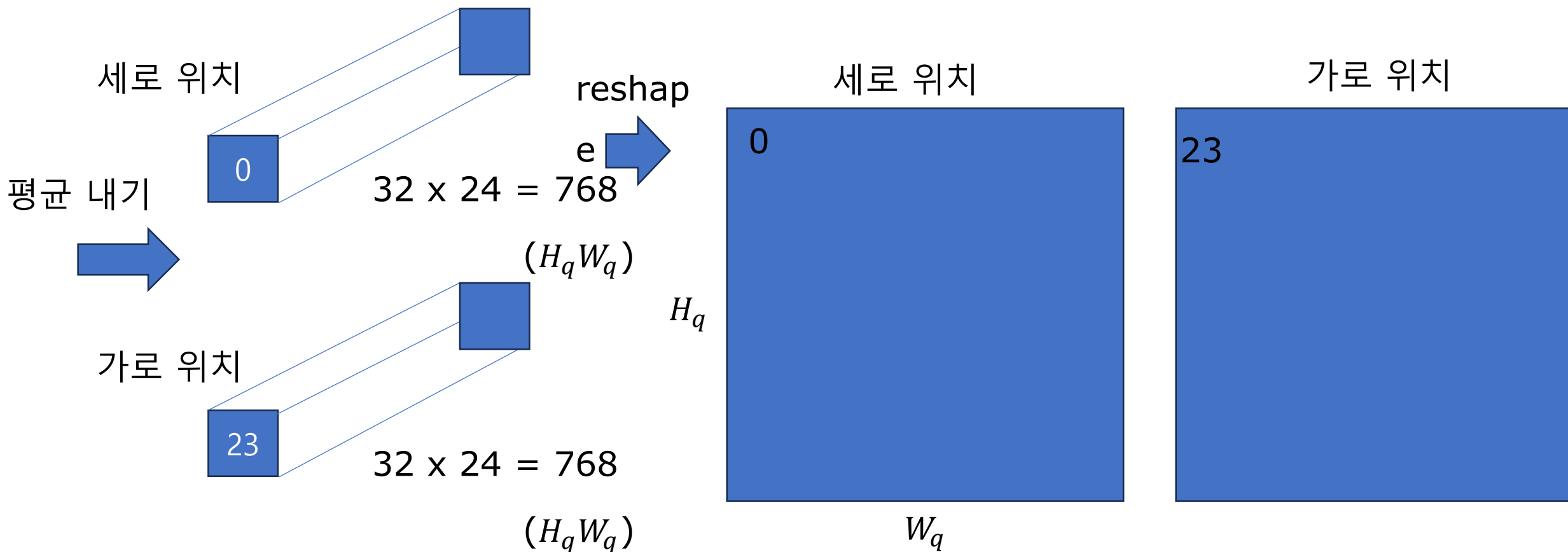
ATV Loss in StableVITON

- 직관적인 해석?
 - 지금까지 따라온 예시 값들은 잘 된 경우이고, 만약에 학습이 덜 된 상태라면 다음과 같이 될 수 있다.



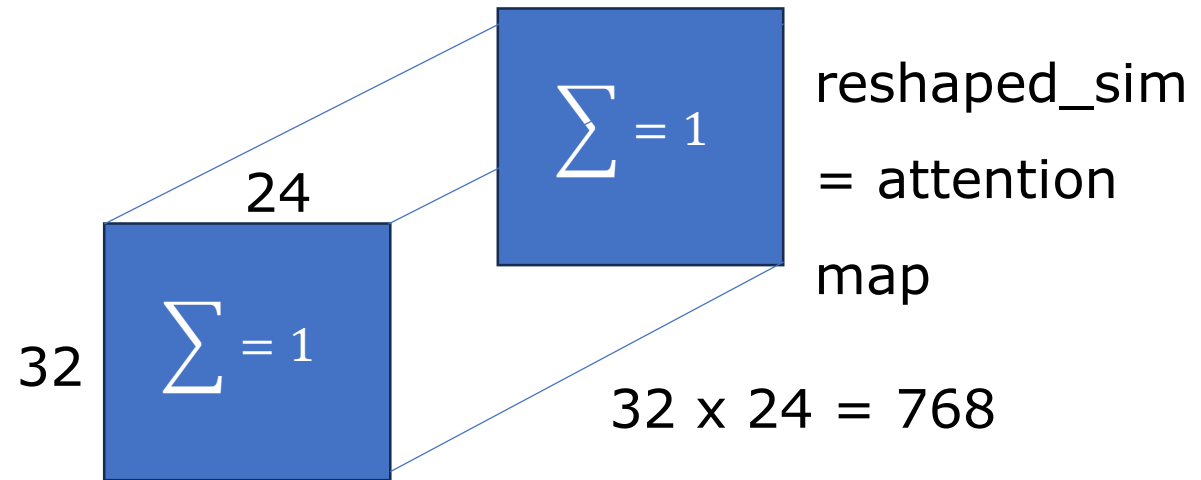
ATV Loss in StableVITON

- 직관적인 해석?
 - 즉, 첫번째 generated patch에 대한 attention score의 중심 좌표는 (0,23)이 나온다.
 - 그렇기에, 너무 학습이 안된 모델에는 이 loss를 적용시키면 오히려 역효과가 날거다.



ATV Loss in StableVITON

- ATV loss: Center coordinate map
 - mH, mh: 32
 - mW, mw: 24
 - h (head): 8



```
mask1 = attn_mask_resize(mask1, mH, mW) # [BS x H x W]
reshaped_sim = sim.reshape(-1, h, mH*mW, mh, mw).mean(dim=1) # [BS, h, HW, h, w] -> [BS, HW, h, w]
mask1_repeat = mask1
h_linspace = torch.linspace(0, mh-1, mh, device=sim.device)
w_linspace = torch.linspace(0, mw-1, mw, device=sim.device)
grid_h, grid_w = torch.meshgrid(h_linspace, w_linspace)
grid_hw = torch.stack([grid_h, grid_w]) # [2, 32, 24]

# [BS, 768(HW), 1, 32(h), 24(w)] element-wise [1, 1, 2, 32, 24]
weighted_grid_hw = reshaped_sim.unsqueeze(2) * grid_hw.unsqueeze(0).unsqueeze(0) # [b HW 2 h w]
weighted_centered_grid_hw = weighted_grid_hw.sum((-2, -1)) # [b HW 2]

tv_loss = get_tvloss(weighted_centered_grid_hw, ~mask1_repeat, ch=mh, cw=mw)
attn_loss = tv_loss * 0.001
```

Entropy

- Attention map 에 warped_cloth_mask 를 곱해서, generated image 의 의미 있는 영역에 대해
확률 값이 고르게 분포되어 있는게 아니라 한 원소가 dominate 하도록 장려

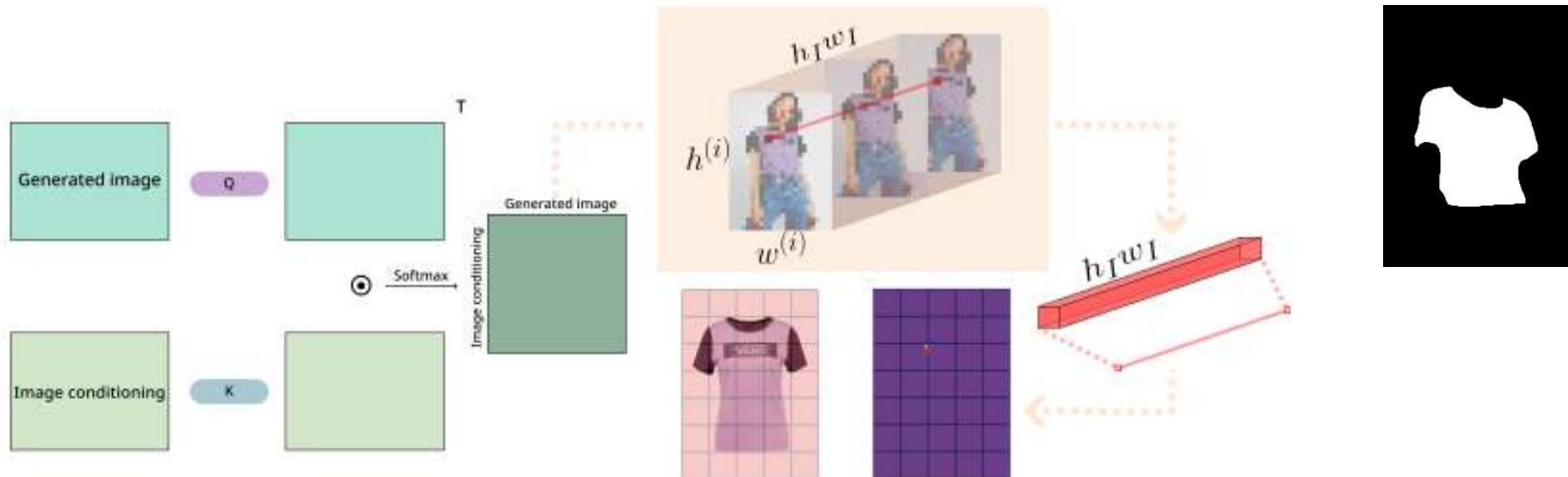


Figure 2: Visualization of image conditioning using IAM. We identify which image conditioning patches a patch of the generated image referenced.

Results

