Relation Networks for Object Detection

(2018 CVPR Oral paper)

Abstract

(1) Although it is well believed for years that modeling relations between objects would help object recognition, there has not been evidence that the idea is working in the deep learning era
: 객체 인식에서 관계를 모델링 하는게 도움이 될 것 같았지만, 딥러닝 분야에서 아직 증면한 사례가 없다

(2) It processes a set of objects simultaneously through interaction between their appearance feature and geometry
: Appearance feature와 geometry 간의 상호 작용을 통해 일련의 객체를 동시에 처리해서, 관계를 모델링

(3) It is shown effective on improving object recognition and duplicate removal steps in the modern object detection pipeline.
: object recognition과 NMS 에서 효과가 있음을 증명

Introduction

(1) Given a sparse set of region proposals, object classification and bounding box regression are performed on each proposal individually.
: 기존의 모델들은 객체를 개별적으로 인식

(2) An attention module can effect an individual element (e.g., a word in the target sentence in machine translation) by aggregating information (or features) from a set of elements. The aggregation weights are automatically learnt, driven by the task goal
: attention 모듈은 요소 집합에서 정보(또는 특징)를 취합하여 개별 요소에 영향을 미칠 수 있다. 통합 가중치는 작업 목표에 따라 자동으로 학습

(3) An attention module can model dependency between the elements, without making excessive assumptions on their locations and feature distributions.
: attention 모듈은 위치 및 형상 분포를 지나치게 가정하지 않고 요소 간의 의존성을 모델링

(4) In this work, for the first time we propose an adapted attention module for object detection.
이 작업에서, 우리는 처음으로 물체 감지를 위한 adapted attention 모듈을 제안

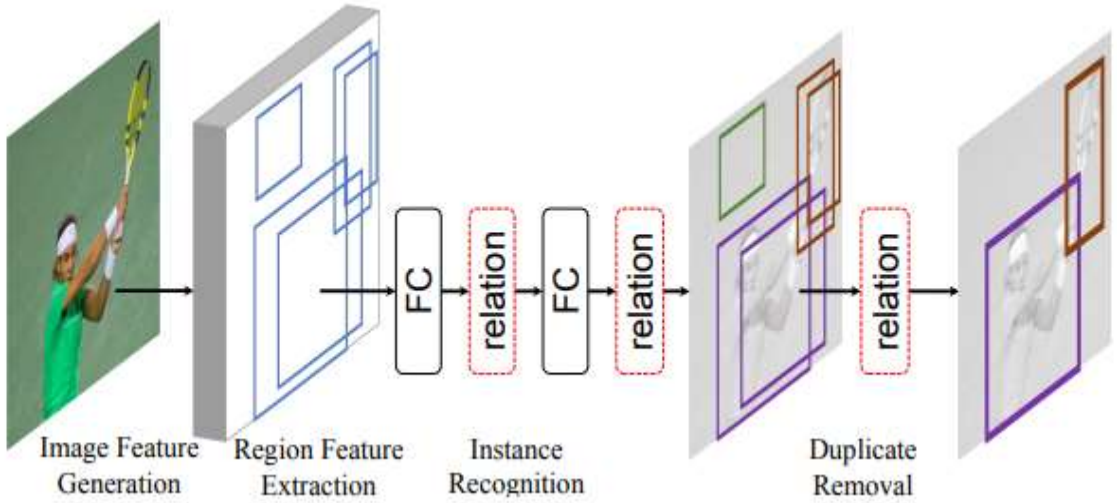
Introduction

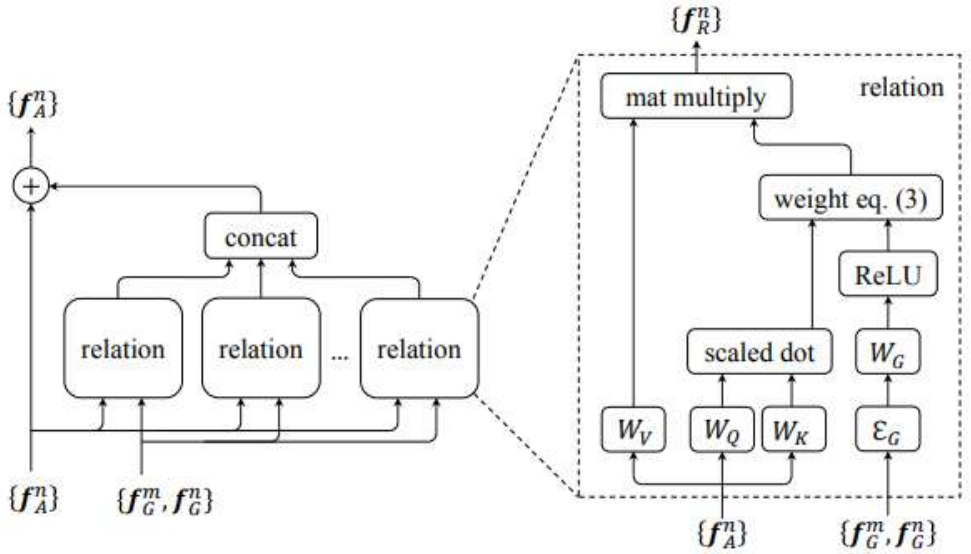
(5) An apparent distinction is that the primitive elements are objects instead of words. The objects have 2D spatial arrangement and variations in scale/aspect ratio. Their locations, or geometric features in a general sense, play a more complex and important role than the word location in an 1D sentence
: 명백한 차이점은 기본 요소가 단어 대신 객체라는 것이고, 객체는 2D 공간 배열과 스케일 / 가로 세 비율의 변화가 있다. 객체의 위치,기하학적 특징은 1D 문장의 단어 위치보다 더 복잡하고 중요한 역할을 합니다.

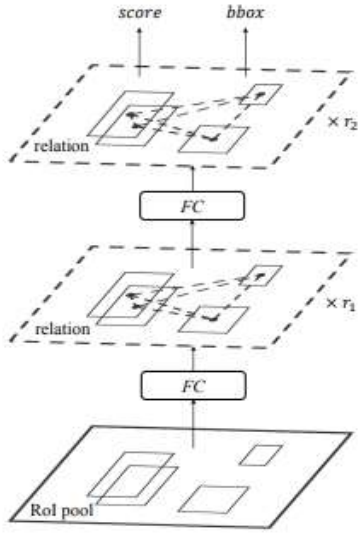
(6) Accordingly, the proposed module extends the original attention weight into two components: the original weight and a new geometric weight. The latter models the spatial relationships between objects and only considers the relative geometry between them, making the module translation invariant, a desirable property for object recognition
: Relation 모듈은 original weights와 새로운 geometric weigh의 두 가지 구성 요소로 확장한다.
후자는 객체 간의 공간적 관계를 모델링하고 그 사이의 상대적 기하학적 구조만 고려하므로(making the module translation invariant,) 모듈 변환이 객체 인식에 좋은 영향을 미친다.

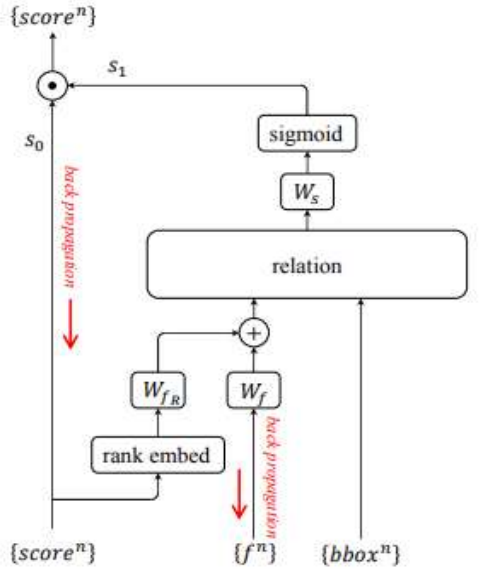
(7) It takes variable number of inputs, runs in parallel (as opposed to sequential relation modeling is fully differentiable and is in-place
: 입력의 가변 수를 사용하고, 병렬로 실행하며(순차적 관계 모델링과는 반대로), 완전히 다르며, inplace 하다(no dimension change between input and output)

Object Relation Module

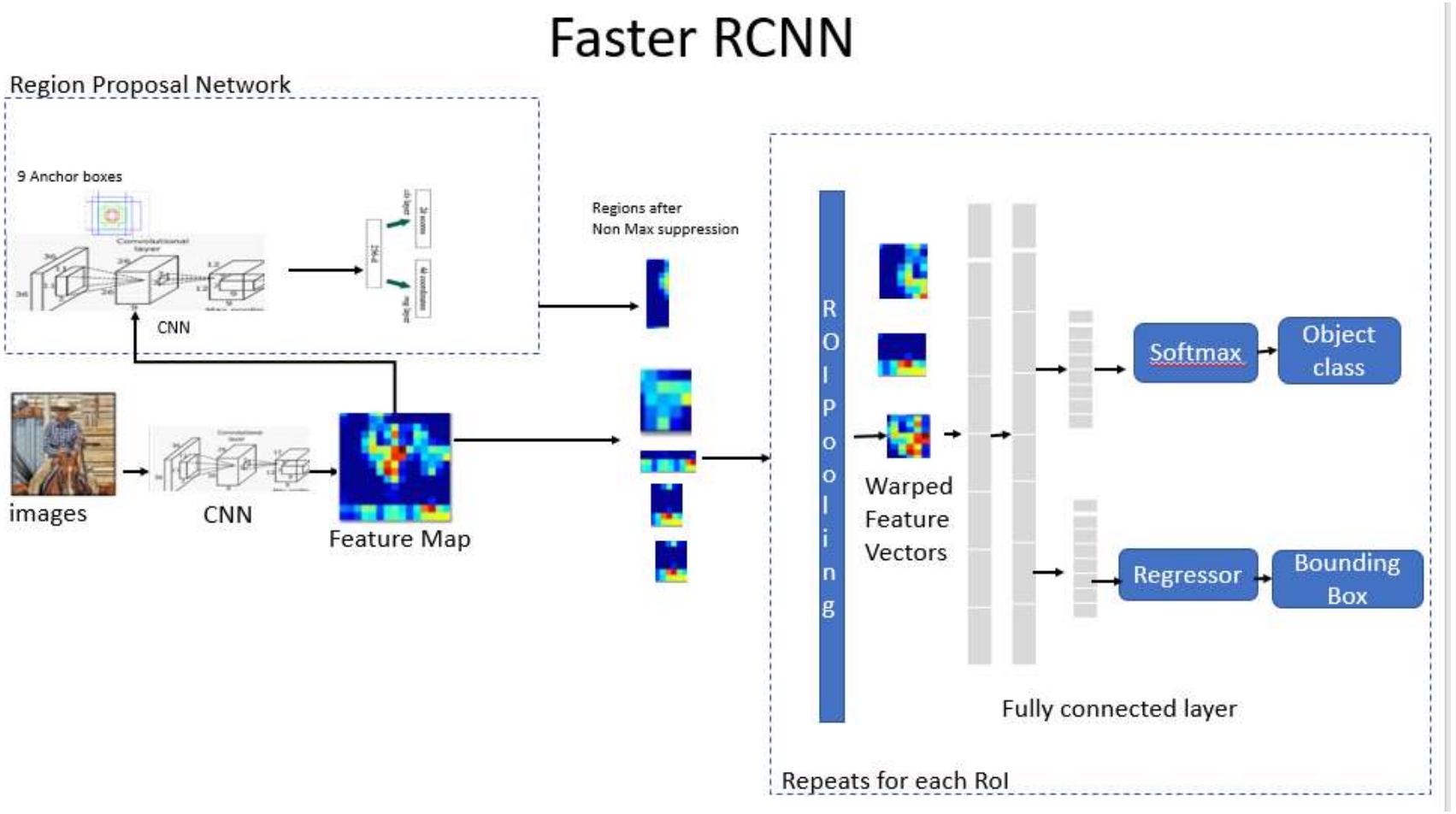








Object Relation Module



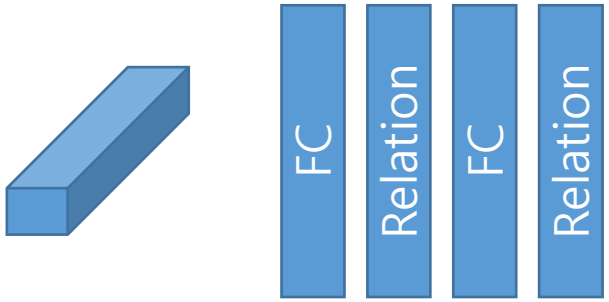
https://medium.com/datadriveninvestor/computer-vision-a-journey-from-cnn-to-mask-r-cnn-and-yolo-1d141eba6e04

Input	
Conv1	7x7, 64, stride2
Layer2_x (3blocks)	3x3 max pool, stride2
	1*1, 64 3x3, 64 1x1, 256
Layer4_x (4blocks)	1*1, 128 3x3, 128 1x1, 512
Layer6_x (23blocks)	1*1, 256 3x3, 256 1x1, 1024

rpn_bbox	rpn_cls
3x3, 512, same	
1x1, 4x9	1x1, 2x9

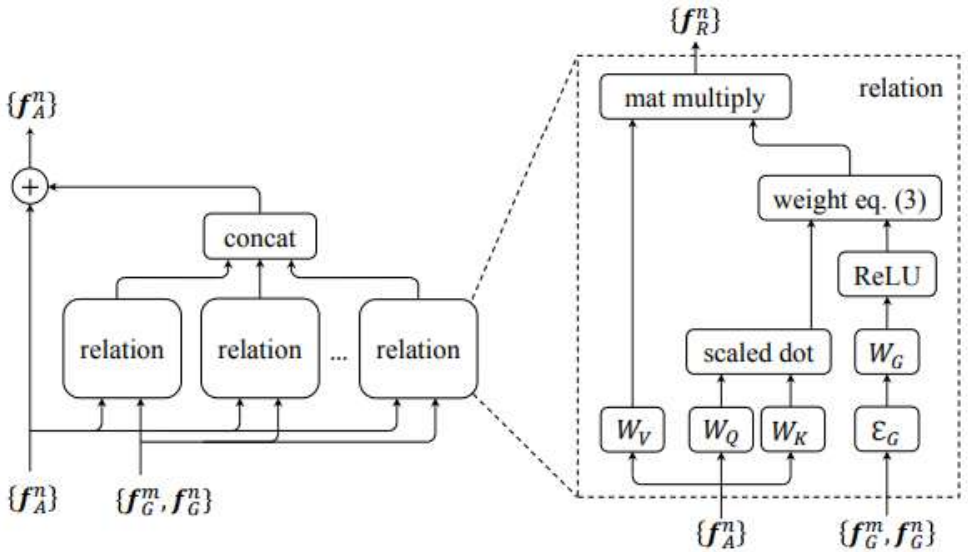
head_bbox	head_cls
RoI Pooling 2x2 max pooling, stride 2	
Flatten	
Avg pooling	

Object Relation Module



fa : appearance feature (1024)
fg: geometry feature (4)
Nr : 16

$$\omega^{mn} = \frac{\omega_G^{mn} \cdot \exp(\omega_A^{mn})}{\sum_k \omega_G^{kn} \cdot \exp(\omega_A^{kn})}$$



0.5	0.4	0.3
0.1	0.6	0.9
0.3	0.7	0.8

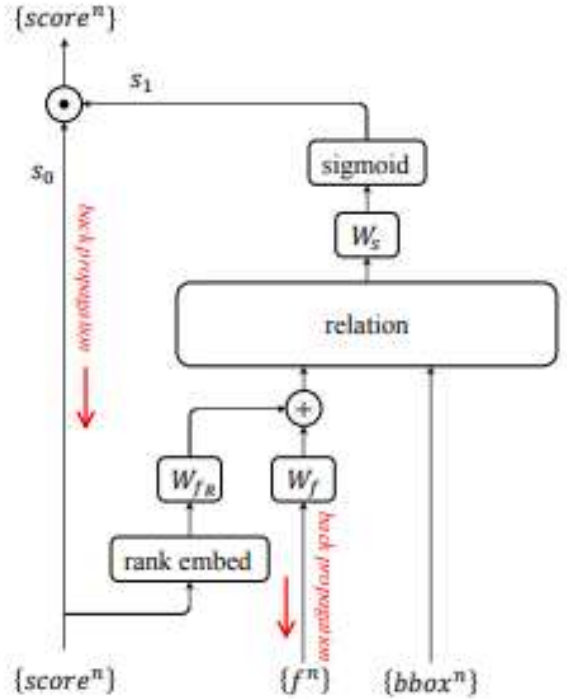
0.1	0.2	0.3

64	64	64
W_v, W_q, W_k		
1024	1024	1024

64	64	64
Position Embedding		
4	4	4

새 사람 사과

Object Relation Module



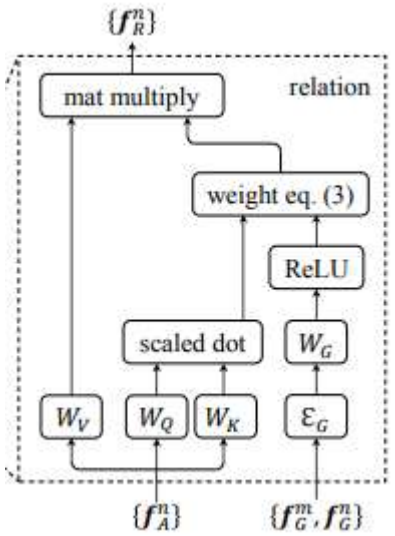
1. RoI 중에서, background가 아니고 score 값으로 내림 찬순 정렬 (ex 200)
2. 200 * 1024개 만큼 rank embedding
3. Rank embedding FC를 통해 128(W_{fR})
4. Sorting된 1024 feature들을 Fc를 통해 128(W_f)
5. 3+4와 sorting된 bbox를 relation 모듈에 태움
6. Relation 결과를 output이 1인 FC에 태우고 sigmoid
7. Sorting된 prob과 결과를 곱함

S1=1(correct), S1=0(duplicate)

Experiments

2fc baseline	*(a): usage of geometric feature*			*(b): number of relations N_r*						*(c): number of relation modules $\{r_1, r_2\}$*				
	none	unary	ours*	1	2	4	8	16*	32	$\{1,0\}$	$\{0,1\}$	$\{1,1\}$*	$\{2,2\}$	$\{4,4\}$
29.6	30.3	31.1	**31.9**	30.5	30.6	31.3	31.7	**31.9**	31.7	31.7	31.4	31.9	32.5	**32.8**

Table 1. Ablation study of relation module structure and parameters (* for default). mAP@all is reported.



1) none : geometric weight = 1
   (bbox 좌표에 대한 정보를 이용 안함)
   unary : f_G is embedded into a high-dimension and added onto fA to form the new appearance feature
   (bbox 좌표를 f_A로 add)
2) Relation Nr parameter에 대한 실험
3) Relation module number에 대한 실험

$$\omega^{mn} = \frac{\omega_G^{mn} \cdot \exp(\omega_A^{mn})}{\sum_k \omega_G^{kn} \cdot \exp(\omega_A^{kn})}.$$

$$\omega_G^{mn} = \max\{0, W_G \cdot \mathcal{E}_G(\mathbf{f}_G^m, \mathbf{f}_G^n)\}.$$

Experiments

head	mAP	mAP_{50}	mAP_{75}	# params	# FLOPS
(a) 2fc (1024)	29.6	50.9	30.1	38.0M	80.2B
(b) 2fc (1432)	29.7	50.3	30.2	44.1M	82.0B
(c) 3fc (1024)	29.0	49.4	29.6	39.0M	80.5B
(d) 2fc+res $\{r_1, r_2\}$=$\{1, 1\}$	29.9	50.6	30.5	44.0M	82.1B
(e) 2fc (1024) + global	29.6	50.3	30.8	38.2M	82.2B
(f) 2fc+RM $\{r_1, r_2\}$=$\{1, 1\}$	**31.9**	53.7	33.1	44.0M	82.6B
(g) 2fc (1024) + 2×	30.4	51.7	31.4	50.2M	83.8B
(h) 2fc+2×+RM $\{r_1, r_2\}$=$\{1, 1\}$	**32.5**	54.3	34.1	56.2M	86.2B
(i) 2fc+res $\{r_1, r_2\}$=$\{2, 2\}$	29.8	50.5	30.5	50.0M	84.0B
(j) 2fc+RM $\{r_1, r_2\}$=$\{2, 2\}$	**32.5**	54.0	33.8	50.0M	84.9B

(a) 2fc + 1024 feature map
(b) 2fc + 1432 feature map (+0.1 mAP) - wider
(c) 3fc + 1024 feature map (-0.6 mAP) - deeper
(d) residual blocks (+0.3 mAP)
(e) global average pooled (no improvement)
(f) relation Module(+2.3 mAP)
(g) 2× larger RoI (+0.8mAP)
(h) 2× larger RoI + RM (+2.9mAP)
(i) (d) x 2 (+0.2 mAP)
(j) (f) x 2 (+2.9 mAP)

NMS	ours	rank f_R		appearance f	geometric *bbox*	
	$\{f_R, f, bbox\}$	*none*	s_0	*none*	*none*	*unary*
29.6	**30.3**	26.6	28.3	29.9	28.1	28.2

Table 3. Ablation study of input features for duplicate removal network (*none* indicates without such feature).

1) rank feature is not used, mAP drops to 26.6.
2) score is embedded to 128-d, mAP drops to 28.3.
3) f_A is not used, mAP slightly drops to 29.9.
4) geometric feature is not used, mAP drops to 28.1.
5) Bbox add to f_A, mAP drops to 28.2.
These results suggest that rank feature is most crucial for final accuracy.

Experiments

method	parameters	mAP	mAP_{50}	mAP_{75}
NMS	$N_t = 0.3$	29.0	51.4	29.4
NMS	$N_t = 0.4$	29.4	**52.1**	29.5
NMS	$N_t = 0.5$	29.6	51.9	29.7
NMS	$N_t = 0.6$	**29.6**	50.9	30.1
NMS	$N_t = 0.7$	28.4	46.6	**30.7**
SoftNMS	$\sigma = 0.2$	30.0	**52.3**	30.5
SoftNMS	$\sigma = 0.4$	30.2	51.7	31.3
SoftNMS	$\sigma = 0.6$	**30.2**	50.9	31.6
SoftNMS	$\sigma = 0.8$	29.9	49.9	**31.6**
SoftNMS	$\sigma = 1.0$	29.7	49.7	31.6
ours	$\eta = 0.5$	30.3	**51.9**	31.5
ours	$\eta = 0.75$	30.1	49.0	**32.7**
ours	$\eta \in [0.5, 0.9]$	**30.5**	50.2	32.4
ours (e2e)	$\eta \in [0.5, 0.9]$	**31.0**	51.4	32.8

Table 4. Comparison of NMS methods and our approach (Section 4.3). Last row uses end-to-end training (Section 4.4).

backbone	test set	mAP	mAP_{50}	mAP_{75}	#. params	FLOPS
faster RCNN [44]	*minival*	32.2→34.7→**35.2**	52.9→55.3→**55.8**	34.2→37.2→**38.2**	58.3M→64.3M→64.6M	122.2B→124.6B→124.9B
	test-dev	32.7→35.2→**35.4**	53.6→**56.2**→56.1	34.7→37.8→**38.5**		
FPN [37]	*minival*	36.8→38.1→**38.8**	57.8→59.5→**60.3**	40.7→41.8→**42.9**	56.4M→62.4M→62.8M	145.8B→157.8B→158.2B
	test-dev	37.2→38.3→**38.9**	58.2→59.9→**60.5**	41.4→42.3→**43.3**		
DCN [11]	*minival*	37.5→38.1→**38.5**	57.3→57.8→**57.8**	41.0→41.3→**42.0**	60.5M→66.5M→66.8M	125.0B→127.4B→127.7B
	test-dev	38.1→38.8→**39.0**	58.1→**58.7**→58.6	41.6→42.4→**42.9**		

Table 5. Improvement (2fc head+SoftNMS [5], 2fc+RM head+SoftNMS and 2fc+RM head+e2e from left to right connected by →) in state-of-the-art systems on COCO *minival* and *test-dev*. Online hard example mining (OHEM) [47] is adopted. Also note that the strong SoftNMS method ($\sigma = 0.6$) is used for duplicate removal in non-e2e approaches.

N_t : IoU threshold
σ : normalizing parameter
η : relation module parameter

- 各 Backbone model 에서 method 별 mAP 결과
1) 2fc + SoftNMS
2) 2fc + RM + SoftNMS
3) 2fc + RM + e2e