Supplementary

Guanyu Cai[1,2]*, Jun Zhang[2], Xinyang Jiang[3]†, Yifei Gong[2],
Lianghua He[1], Fufu Yu[2], Pai Peng[2], Xiaowei Guo[2], Feiyue Huang[2], Xing Sun[2]†
Tongji University[1], Tencent Youtu Lab[2], Microsoft Research[3]
{caiguanyu,Helianghua}@tongji.edu.cn,xinyangjiang@microsoft.com,pengpai_sh@163.com
{bobbyjzhang,yifeigong,fufuyu,scorpioguo,garyhuang,winfredsun}@tencent.com

1. Network Implementation Details

In this section, we describe implementation details of the parameterized components in Ask&Confirm: Text Encoder, Image Encoder, policy net, and value net.

Text Encoder. We map the natural language to a 256-dimensional vector space. Given a sentence T that contains n words, we represent the i th word in it with a one-hot vector showing the index of the word in a vocabulary and then embed the word into a 300-dimensional vector x_i through an embedding matrix W_e. Then, we use a one-layer unidirectional GRU to map the vector to the final textual feature along with the sentence context. The GRU reads the sentence T from 1 to n th word and obtains the final textual feature x^T:

$$x^T = GRU(x_i), i \in [1, n] \tag{1}$$

We do not use a bidirectional GRU in this work because the performance between them is close according to our experimental results.

Image Encoder. Given an image I, we aim to map it to a set of 256-dimensional vectors $X^I = \{x_1^I, x_2^I, ..., x_k^I\}, k = 36$ where each vector encode a region and predict a set of objects $A = \{a_1, a_2, ..., a_j\}$ in an image. We refer to detection of salient regions as bottom-up attention [1] and implement it with a Faster-RCNN [5]. We adopt the Faster-RCNN whose backbone is a ResNet-101 [2] pretrained by Anderson et al. [1] on Visual Genome [3]. For each region i, f_i is defined as the mean-pooled feature from this region and the dimension of f_i is 2048. To get a 256-dimensional vector as textual vectors, we add an two-layer MLP to transform f_i to x_i^I:

$$x_i^I = MLP(f_i) \tag{2}$$

As for predicting objects, the original model predicts attribute classes and instance classes together to learn feature representations with rich semantic meaning. However, in our Ask&Confirm, we just need objects in an image. Hence, we re-train a two-layer MLP to predict the objects in I. After obtaining X^I, we concatenate all vectors into a 36×256-dimensional vector X_1^I and use the MLP to predict every object's probability of being in I. The architecture of the re-trained MLP is shown in Table 1.

Type	Weight shape	Input size
Fc	2048×256	N×2048
Fc	256×256	N×256
Fc	9216× 256	N×9216
Fc	256× 1601	N×256

Table 1: The architecture of MLP that predicts the objects in an image. N denotes the batchsize and Fc denotes the fully-connected layer.

Type	Weight shape	Input size
Fc	3202× 256	N×3202
Tanh	-	N×256
Fc	256×256	N×256
Tanh	-	N×256
Fc	256× 1601	N×256
Softmax	-	N×1601

Table 2: The architecture of policy net. Tanh denotes the hyperbolic tangent function and Softmax denotes the softmax function.

Type	Weight shape	Input size
Fc	3202× 256	N×3202
Tanh	-	N×256
Fc	256×1	N×256

Table 3: The architecture of value net.

Policy Net. Given a state $s \in \mathbb{R}^{3202}$. The policy net

*Work done during internship at Youtu Lab
†Corresponding author: Xinyang Jiang, Xing Sun

π outputs a 1601-dimensional vector as the object sample distribution. During training, we apply a stochastic sampling to choose objects to users while in the testing period, a greedy sampling is applied. The architecture of π is shown in Table 2.

Value Net. Given a state $s \in \mathbb{R}^{3202}$. The value net V outputs a scalar that estimates the real advantage returned by the interactive agent. According to [6], estimating the advantage is helpful to reduce the variance of reinforcement learning. The architecture of V is shown in Table 3.

2. Implementation Details of Partial Query v.s. Partial Query + Objects

To demonstrate that objects in an image are discriminative enough to distinguish different images, we conduct an experiment, i.e. partial query v.s. partial query + objects, to compare two types of queries: partial query and supplement partial query with the name of the objects. The experiment is evaluated on Visual Genome [3].

In detail, for each image i that includes several captions $Q = \{q_n\}_{n=1}^{N_Q}$ to describe it, we randomly choose one caption q_n as the partial query. As for the additional objects, we use an object detector [1] pretrained on Visual Genome [3] to detect all objects $A = \{a_n\}_{n=1}^{N_A}$ contained in each image. These objects' names are regarded as additional queries. For example, if an initial query q, i.e. "a man is surfing", is chosen to retrieve its corresponding image i and the detector detects all objects A, i.e. "man", "sea" and "surfboard", in the target image, these words of objects are regarded as three individual queries appended to the initial query. Thus, the new queries includes four query: "a man is surfing", "man", "sea" and "surfboard". As a result, the new query adds more discriminative information to retrieve the target image.

3. Pseudo Code

To describe our Ask&Confirm in more detail, we give the pseudo code of the whole workflow of Ask&Confirm as shown in Algorithm 1.

Algorithm 1 The whole workflow of Ask&Confirm

Initialize Text Encoder TE and Image Encoder IE
Initialize policy parameters ϕ_π and value parameters ϕ_V
Input: $I = \{i_n\}_{n=1}^{N}$: the whole gallery images
for episode=1,M **do**
  Input: i_*: the target image
  Input: $Q_1 = \{q_n\}_{n=1}^{N_Q^1}$: a set of input partial queries
  for t=1, T **do**
    for n=1, N_Q^t **do**
      $x_n^T = TE(q_n)$
    end for
    Obtain textual features $X_t^T = \{x_n^T\}_{n=1}^{N_Q^T}$
    for n=1, N **do**
      $(x_n^I, A_n) = IE(i_n)$
      Compute similarity $S_{t,n}(X_t^T, x_n^I)$
    end for
    Question: Object candidates: $A_t = \{a_n\}_{n=1}^{N_A}$
    Feedback: Positive objects: $A_t^p = \{a_n^p\}_{n=1}^{N_A^p}$
    Feedback: Negative objects: $A_t^q = \{a_n^q\}_{n=1}^{N_A^q}$
    for n=1, N **do**
      Refine $S_{t,n} = S_{t,n} \times 0.9, if\ A_n \cap A_t^q \neq \emptyset$
    end for
    Output: $i_t = \underset{i_n}{\operatorname{argmax}}\, S_{t,n}$
    Update queries $Q_{t+1} = Q_t \cup A_t^p$
  end for
  if episode $\%N_s == 0$ **then**
    Collect a set of episode
    Run PPO to optimize ϕ_π and ϕ_V
  end if
end for

4. Details of each user

In this section, we give an detailed description of the user study. For the user selection, we follow the metric in Drill Down. An expert user (male) familiar with interactive retrieval, and three novice users (1 female+2 male) are selected. They are volunteering postgraduates. For fair comparison, users are blind to these methods. After showing the target image for 5 sec, a user is asked to retrieve by interacting with the retrieval system. For AC, top-10 retrieved images and object candidates are shown to the user per turn as hints, and the user confirms the presence of objects. Detailed results are shown in Table.4 where Exp denotes the expert user and Nov denotes the novice user.

Furthermore, we conduct the evaluation over iterations of different interactive image retrieval systems to compare their performance. Results are shown in Figure.1. AC obtains similar performance over iterations compared with DD and it outperforms WS with a large margin. As for Mean Rank, AC outperforms other approaches. Considering that

User	AC		DD		WS	
	R@1	R@5	R@1	R@5	R@1	R@5
Exp1	12.0	36.0	12.0	40.0	3.0	16.0
Nov1	8.0	32.0	8.0	38.0	2.0	14.0
Nov2	10.0	32.0	6.0	34.0	2.0	14.0
Nov3	6.0	34.0	8.0	38.0	1.0	12.0

Table 4: Detailed performance of each user.

Method	R@1	R@5	MR	Q	A
AC	8.6	33.9	96.0	1	10
QACohe	7.2	27.6	154.4	1	10
AC	16.8	43.4	70.7	2	5
QACohe	16.1	39.6	113.5	2	5
AC	34.1	61.4	37.8	4	3
QACohe	33.8	58.7	48.4	4	3

Table 5: Performance of AC and QACohe on R@1, R@5 and Mean Rank. Q and A denote the number of queries and actions.

AC costs much less time than DD, we conclude that AC performs the best among the three approaches.

Meanwhile, to give a subjective comparison of user experience, we conduct a post-experiment survey to acquire users' feeling about different interactive retrieval methods. Three of the users (1 Exp+2 Nov) prefer AC and a novice user prefers DD. The reason that they think WS is not user-friendly is the poor performance such that they can hardly find the target image.

5. AC v.s. QACohe

In this section, we give a detailed comparison between AC and QACohe on R@1, R@5 and Mean Rank. Table.5 shows the comparison between AC and QACohe on R@1, R@5 and Mean Rank after 10 turns. Figure.2, 3 and 4 demonstrate the performance of AC and QACohe on R@1, R@5 and Mean Rank. Meanwhile, Standard Deviation (SD) of AC, DD and WS on R@1 is 2.24, 2.18, 0.71. SD on R@5 is 1.66, 2.18, 1.41. SD on R@10 is 1.41, 2.60, 2.60. It is obvious that AC outperforms QACohe in all settings, which demonstrates that our RL-based policy is better than pre-defined policies.

6. Visualizations

In this section, we provide more visualizations of Ask&Confirm based on SCAN [4] to verify the effectiveness of it. We perform Ask&Confirm in three settings: (1) Q1/A10, (2) Q2/A5, and (3) Q4/A3. QK means K queries are given by users in the beginning, and AK means K actions are provided by an agent in each round. In detail, we visualize Ask&Confirm with Q1/A10, Q2/A5, Q4/A3 in Figure 5, 6 and 7, respectively.

References

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086, 2018. 1, 2

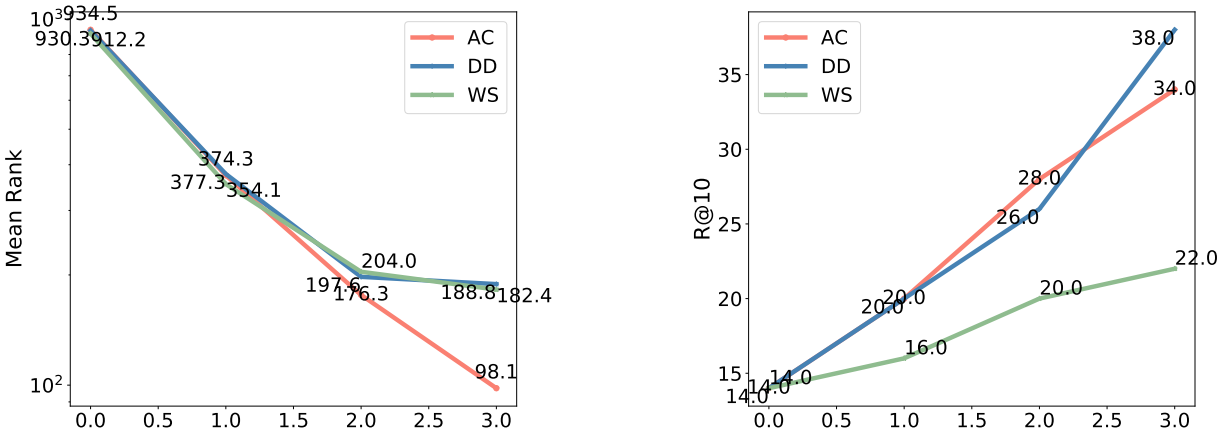
[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 1

[3] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73, 2017. 1, 2

[4] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 201–216, 2018. 3

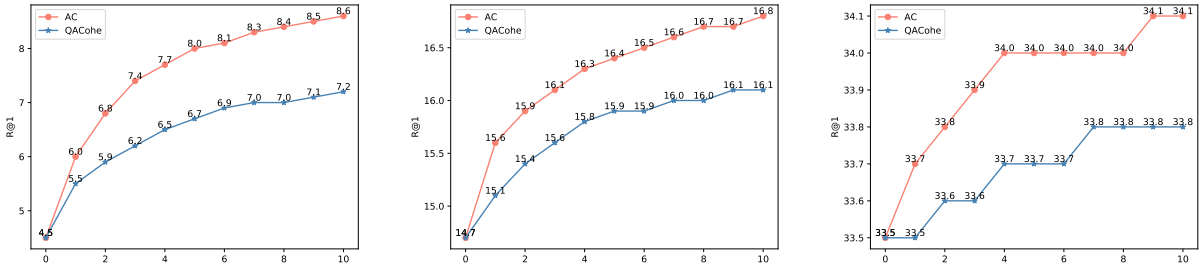
[5] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, pages 91–99, 2015. 1

[6] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 2



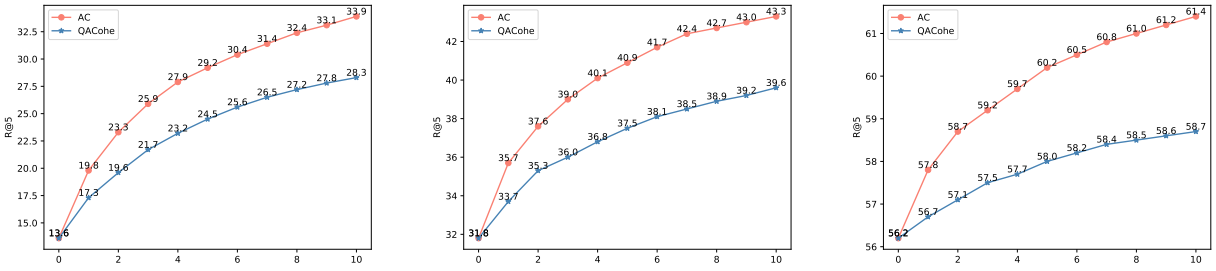
(a) Mean Rank (b) R@10

Figure 1: Mean Rank (lower is better) and R@10 (higher is better) over iterations of different interactive image retrieval systems. Ac denotes Ask&Confirm, DD denotes Drill Down and WS denotes WhittleSearch.



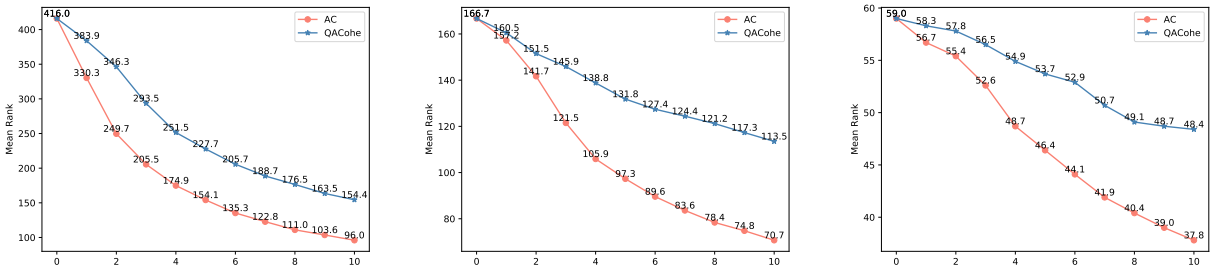
(a) Query 1/Action 10 (b) Query 2/Action 5 (c) Query 4/Action 3

Figure 2: Results of AC and QACohe on R@1.The horizontal axis represents the query turn.



(a) Query 1/Action 10 (b) Query 2/Action 5 (c) Query 4/Action 3

Figure 3: Results of AC and QACohe on R@5.The horizontal axis represents the query turn.



(a) Query 1/Action 10 (b) Query 2/Action 5 (c) Query 4/Action 3

Figure 4: Results of AC and QACohe on Mean Rank.The horizontal axis represents the query turn.

child in a stroller

Rank: 1173

Round 1: Window, Sky, Person, Building, Man, Head, Shirt, Hand, Wall, Grass **Rank: 119**

Round 3: Shadow, People, Face, Clouds, Line, Leg, Road, Fence, Arm, Olives **Rank: 10**

fourescent porch lighting of an white three story building

Rank: 810

Round 1: Window, Tree, Head, Person, Ground, Man, Shirt, Hair, Wall, Grass **Rank: 318**

Round 3: Door, Road, Pants, Line, Fence, Traffic sign, Car, Front window, Olives, People **Rank: 61**

Round 8: Weeds, Umbrella, Mouth, Cap, Mantle, Hands, Baseboard, Foot, Roof, Clock **Rank: 6**

person on the boat

Rank: 778

Round 1: Man, Shirt, Head, Window, Sky, Person, Hand, Hair, Wall, Grass **Rank: 453**

Round 3: Building, Field, Floor, Hat, Table, Jacket, Light, Chair, People, Trees **Rank: 39**

Round 6: Water, Boy, Cap, Post, Lights, Foot, Sunglasses, Bag, Lady, Tail **Rank: 3**

Figure 5: Visualizations of Ask&Confirm based on SCAN with Query1/Action10.

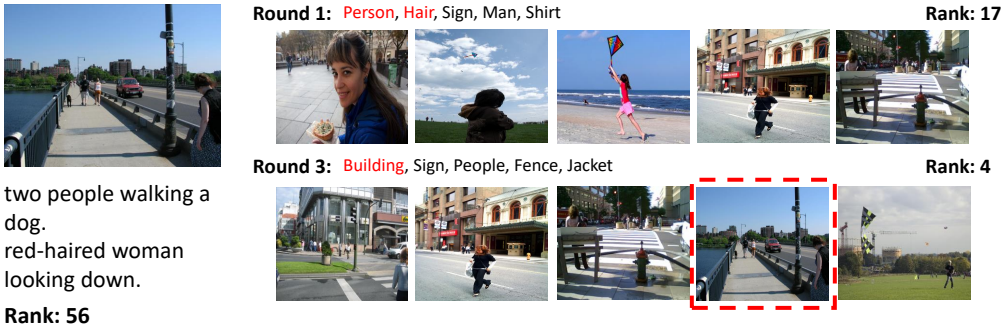


Figure 6: Visualizations of Ask&Confirm based on SCAN with Query2/Action5.

the street is grey.
the sidewalk.
marks on the sidewalk.
a car in the street.

Rank: 153

Round 1: Sky, Ground, Man **Rank: 84**

Round 3: Building, Tree, Grass **Rank: 10**

a very thick grey tree.
large brown mulchy area
around a grey tree.
side of a tan brick
building near a mulch bed.
top of a white umbrella.

Rank: 50

Round 2: Sky, Building, Wall **Rank: 42**

Round 4: Window, Hair, Sign **Rank: 9**

brown bowling ball
going down the lane.
man throwing a
bowling ball.
bowling pins at the end
of the lane.
lanes in the bowling
alley.

Rank: 57

Round 1: Shirt, Man, Person **Rank: 3**

a car on a street.
a window on a building.
this is a city street.
this is a car.

Rank: 58

Round 4: Sign, Head, Window **Rank: 15**

Round 5: Light, Hair, People **Rank: 2**

Figure 7: Visualizations of Ask&Confirm based on SCAN with Query4/Action3.