# Fine-grained Image-text Matching by Cross-modal Hard Aligning Network

Zhengxin Pan[1], Fangyu Wu[2], Bailing Zhang[3]

[1] Zhejiang University, [2] Xi'an Jiaotong-liverpool University,
[3] NingboTech University

Code: https://github.com/ppanzx/CHAN

# Background

- Given collection of images, captions
- Perform retrieval tasks...
  - Image Retrieval
  - Caption Retrieval
- Useful for…
  - Image captioning
  - Visual question answering
  - etc...

Pastry sitting on top of a golden white plate with forks.

The underside of a passenger airliner taking off.

A white jet airliner with blue sky in background.

A large commercial plane flying overhead in the sky.

White jet plane flying in the sky with engines.

The bottom of an airplane flying in the sky.

Examples of bidirectional retrieval
(Figures are copied from [1])

# Background

- Visual Semantic Embedding (VSE) Method
  - Project the holistic image and text into a common embedding space where the overall semantic similarity is measured.
- Fine-grained Matching Method
  - Map visual and linguistic fragments into representation space and then match the embeddings to obtain the overall similarity.

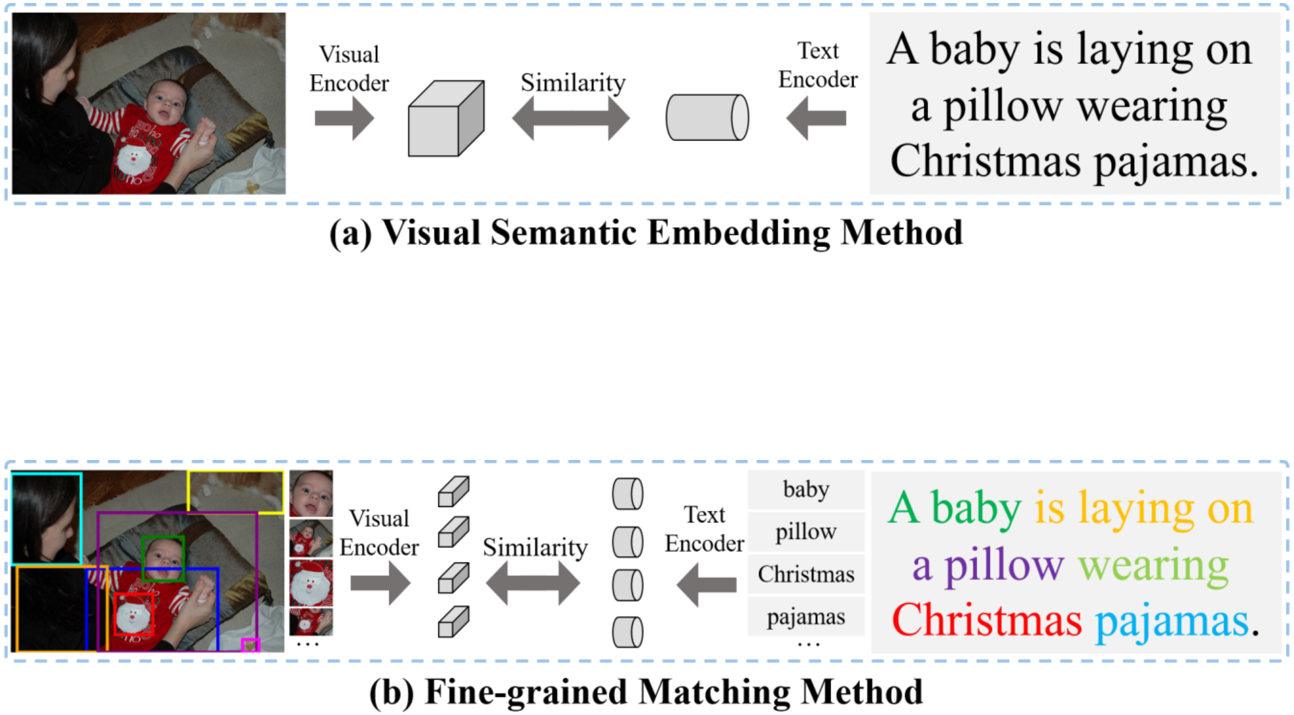



Illustration of VSE and fine-grained matching method.

# Background

- Cross-attention-based fine-grained matching methods (CAM)
    - Utilize cross-attention mechanism to infer the alignment between salient image subregions and text tokens.
- Problems of CAM
    - Sub-optimal accuracy:

        Redundant alignments are detrimental to retrieval accuracy.
    - c:

        Caching cross-attention weights is with a massive cost of memory and time.

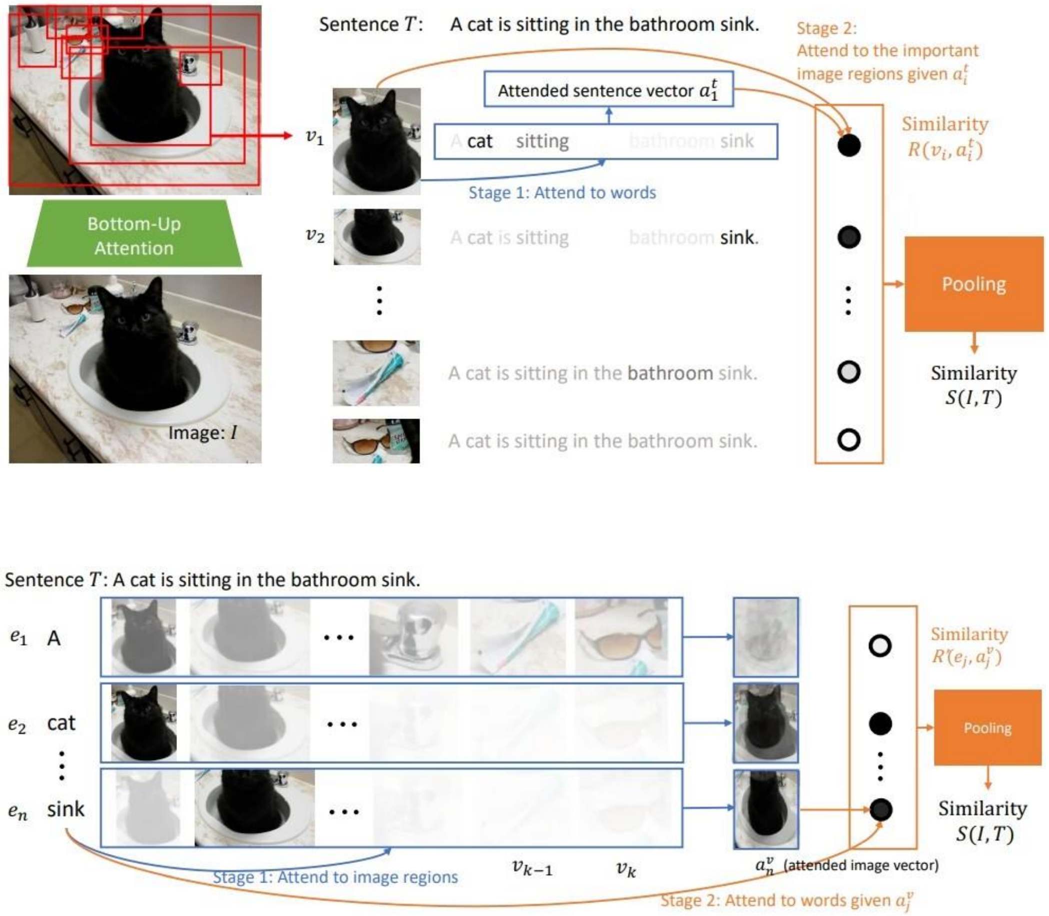
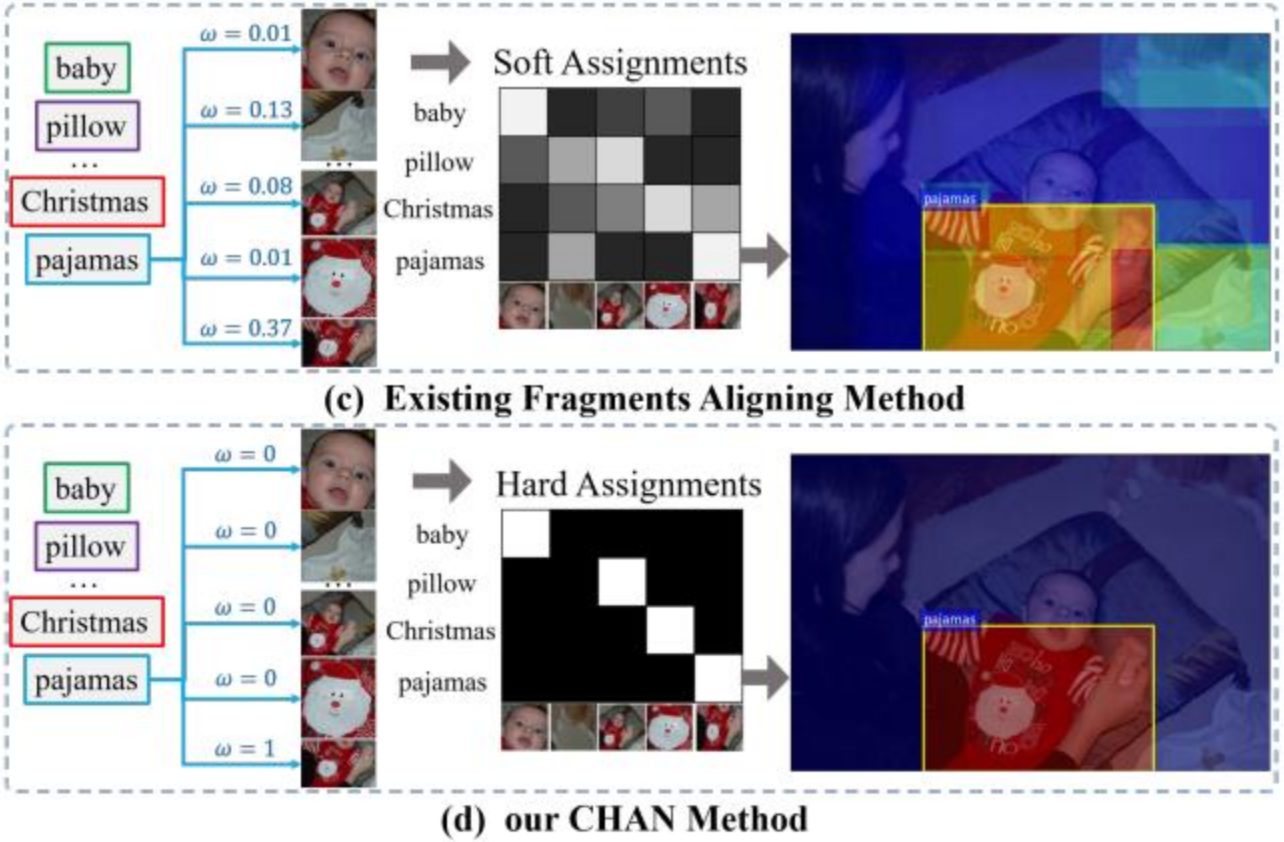



Illustration of CAM.
Figures are copied from [2]

# Our method

- Basic idea
  - A coding framework can well conclude the aligning process. CAM is a special case of soft assignment coding.
- Insight
  - There must exist a sub-region in an image that can best describe every given word in its semantically consistent sentence.
- Our approach
  - We further propose the **C**ross-modal **H**ard **A**ligning **N**etwork (CHAN) based on hard-assignment coding.




Comparison of current CAM and our CHAN.

# Our method

JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

- The fine-grained image-text similarity is actually the weighted sum reconstruction similarity:

$$\boldsymbol{s}(\mathcal{T}, \mathcal{V}) = \frac{1}{\lambda} \log \sum_{i=1}^{L} \exp(\lambda \boldsymbol{s}(\boldsymbol{t}_i, \mathcal{V}))$$

- The reconstruction similarity is the similarity between query $t_i$ and code book $\mathcal{V} = \{v_j\}_{j=1}^{K}$ :

$$\boldsymbol{s}(\boldsymbol{t}_i, \mathcal{V}) = \mathcal{S}(\boldsymbol{t}_i, \hat{\boldsymbol{t}}_i)$$

$$\hat{\boldsymbol{t}}_i = \sum_{j=1}^{K} \omega_{ij} \boldsymbol{v}_j$$

- The retrieval accuracy is highly related with the formulation of coding coefficient
    - soft-assignment coding:

$$\omega_{ij} = \frac{\exp(\boldsymbol{s}_{ij}/\tau)}{\sum_{j=1}^{K} \exp(\boldsymbol{s}_{ij}/\tau)}$$

    - hard-assignment coding

$$\omega_{ij} = \begin{cases} 1\,, & if\ j = \underset{j'=1\cdots K}{arg\ max}(\boldsymbol{s}_{ij'}); \\ 0\,, & otherwise. \end{cases} \quad \Rightarrow \quad \boldsymbol{s}(\boldsymbol{t}_i, \mathcal{V}) = \frac{\boldsymbol{t}_i^\top \hat{\boldsymbol{t}}_i}{\|\boldsymbol{t}_i\| \cdot \|\hat{\boldsymbol{t}}_i\|} = \frac{\boldsymbol{t}_i^\top \boldsymbol{v}_k}{\|\boldsymbol{t}_i\| \cdot \|\boldsymbol{v}_k\|} = \boldsymbol{s}_{ik} = \underset{j=1\cdots K}{max}(\boldsymbol{s}_{ij})$$

# Our method

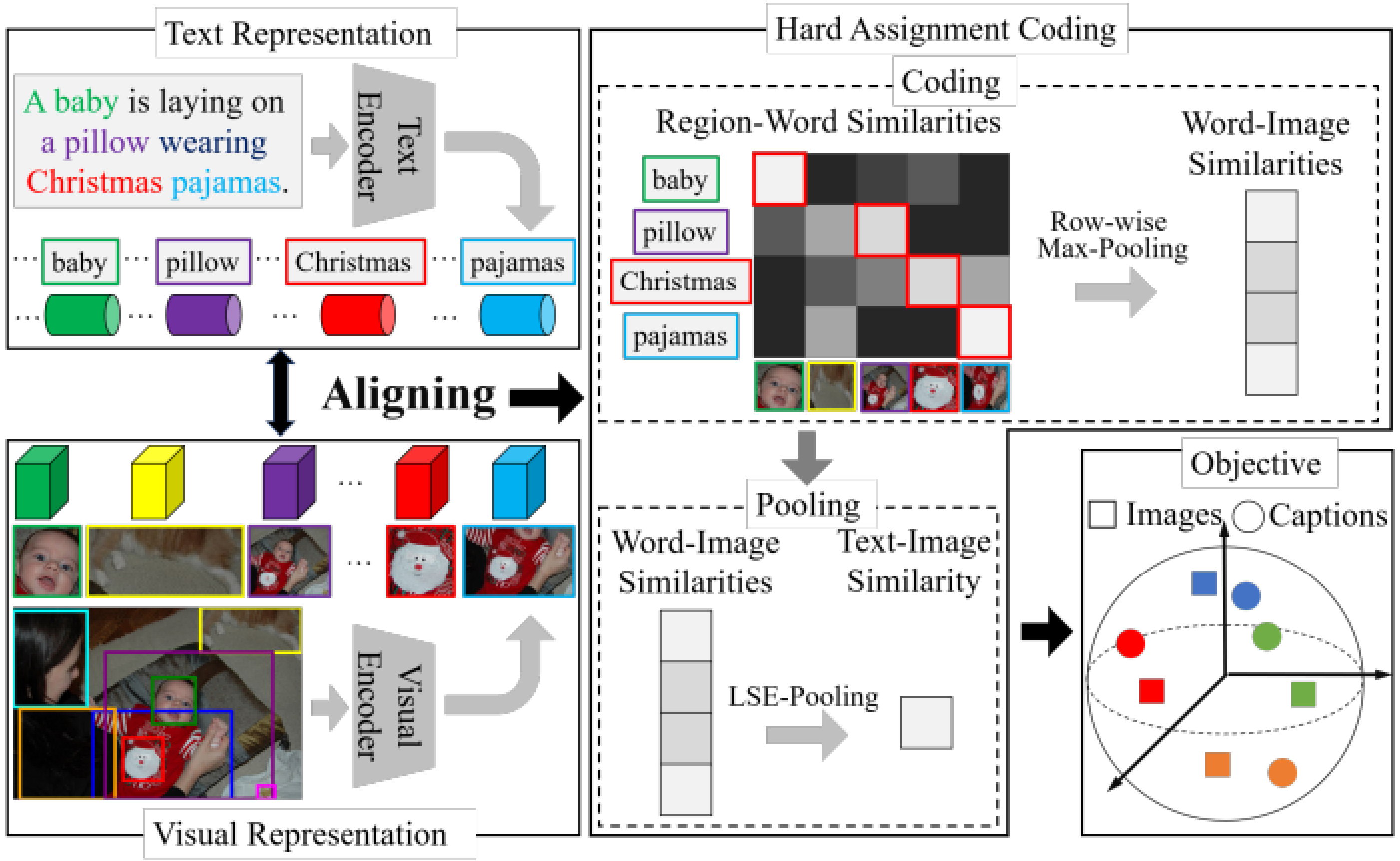
- Modules of CHAN
  - Visual representation

    BUTD-based Faster RCNN
  - Text representation

    Glove+Bi-GRU/pretrained Bert
  - Hard assignment coding

    Row-wise Max-Pooling + LSE Pooling
  - Objective function

    Triplet Loss with hard negative mining



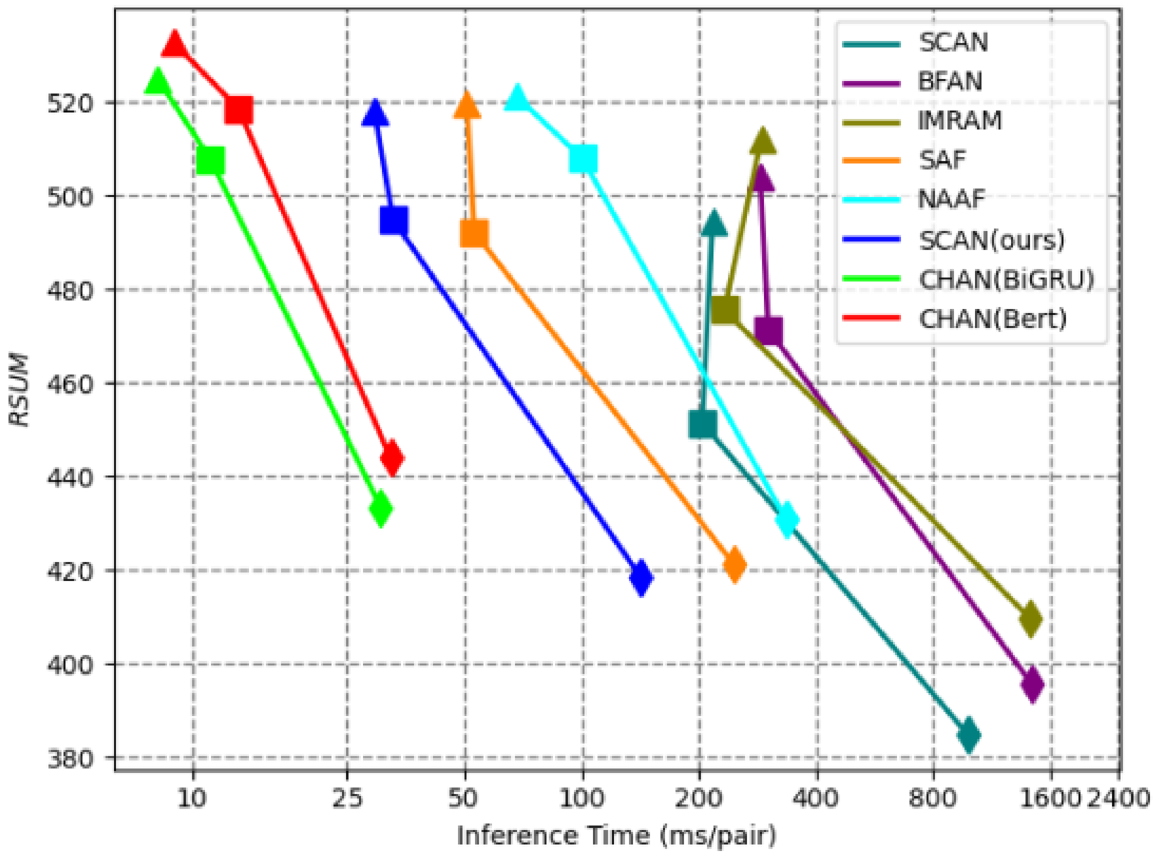

Overview of CHAN

JUNE 18-22, 2023 CVPR VANCOUVER, CANADA

# Experiments

| METHOD | TYPE | COCO 5-fold 1K Test [5] IMG → TEXT R@1 | R@5 | R@10 | TEXT → IMG R@1 | R@5 | R@10 | RSUM |
|---|---|---|---|---|---|---|---|---|
| **ResNet-152 [15] + BiGRU** | | | | | | | | |
| VSE++ [11]2017 | Global | 64.6 | 90.0 | 95.7 | 52.0 | 84.3 | 92.0 | 478.6 |
| VSE∞ [4]2021 | Global | 76.5 | 95.3 | 98.5 | 62.9 | 90.6 | 95.8 | 519.6 |
| **BUTD [1] + BiGRU** | | | | | | | | |
| VSRN⋆ [22]2019 | Fragment | 76.2 | 94.8 | 98.2 | 62.8 | 89.7 | 95.1 | 516.8 |
| VSE∞ [4]2021 | Fragment | 78.5 | 96.0 | 98.7 | 61.7 | 90.3 | 95.6 | 520.8 |
| SCAN⋆ [21]2018 | Aligning | 72.7 | 94.8 | 98.4 | 58.8 | 88.4 | 94.8 | 507.9 |
| IMRAM⋆ [3]2020 | Aligning | 76.7 | 95.6 | 98.5 | 61.7 | 89.1 | 95.0 | 516.6 |
| SGRAF⋆ [10]2021 | Aligning | 79.3 | 96.7 | 98.3 | 64.5 | 90.0 | 95.8 | 524.6 |
| CGMN [6]2022 | Aligning | 76.8 | 95.4 | 98.3 | 63.8 | 90.7 | 95.7 | 520.7 |
| NAAF [46]2022 | Aligning | 78.1 | 96.1 | 98.6 | 63.5 | 89.6 | 95.3 | 521.2 |
| CHAN (ours) | Aligning | 79.7 | 96.7 | 98.7 | 63.8 | 90.4 | 95.8 | **525.0** |
| **BUTD [1] + BERT [9]** | | | | | | | | |
| MMCA [41]2020 | Aligning | 74.8 | 95.6 | 97.7 | 61.6 | 89.8 | 95.2 | 514.7 |
| VSE∞ [4]2021 | Aligning | 79.7 | 96.4 | 98.9 | 64.8 | 91.4 | 96.3 | 527.5 |
| TERAN⋆ [28]2021 | Aligning | 80.2 | 96.6 | 99.0 | 67.0 | 92.2 | 96.9 | 531.9 |
| VSRN++⋆ [23]2022 | Aligning | 77.9 | 96.0 | 98.5 | 64.1 | 91.0 | 96.1 | 523.6 |
| CHAN (ours) | Aligning | 81.4 | 96.9 | 98.9 | 66.5 | 92.1 | 96.7 | **532.6** |

Quantitative Comparison between current SOTAs and our CHAN. CHAN outperforms all of current methods



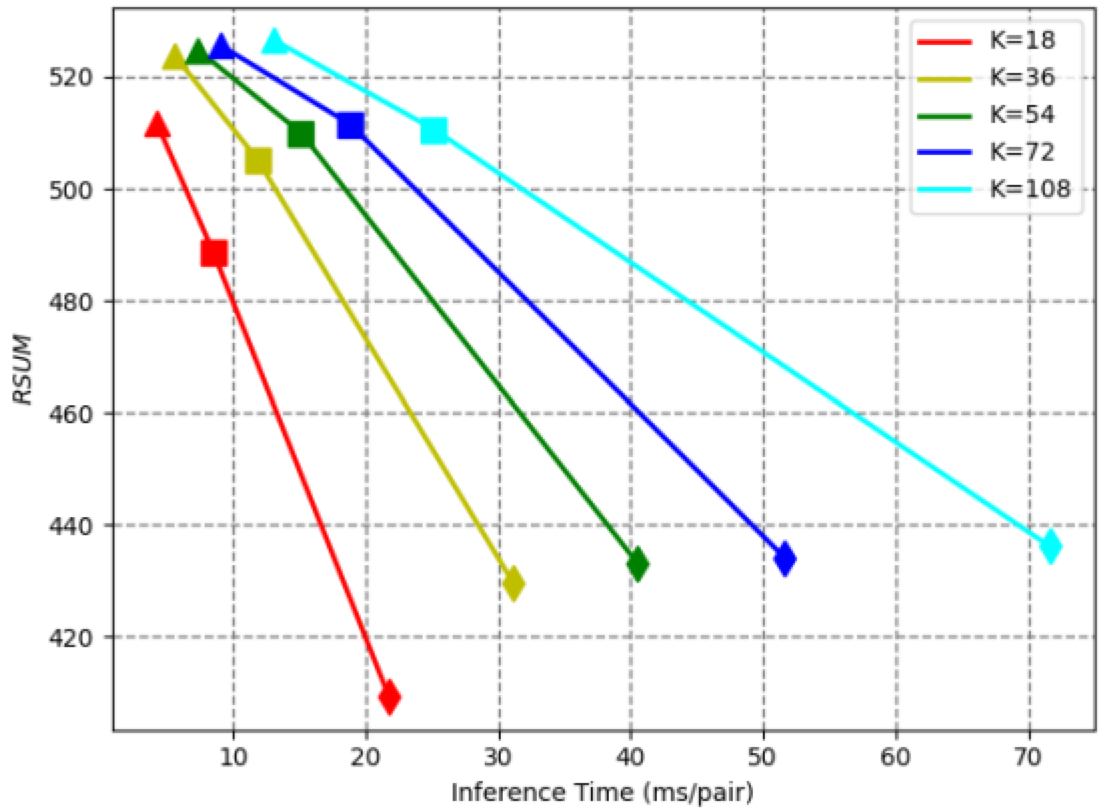

Efficiency Comparison between current SOTAs and our CHAN. CHAN are over 10 times faster than other methods

# Experiments



| METHOD | IMG → TEXT | | | TEXT → IMG | | | RSUM |
|---|---|---|---|---|---|---|---|
| | R@1 | R@5 | R@10 | R@1 | R@5 | R@10 | |
| **Coding Types** | | | | | | | |
| Cross-Attention | 54.8 | 83.7 | 91.2 | 39.6 | 69.0 | 80.3 | 418.6 |
| Visual Codebook | 60.2 | 85.9 | 92.4 | 41.7 | 71.5 | 81.7 | **433.4** |
| Textual Codebook | 48.8 | 80.1 | 88.9 | 35.2 | 66.6 | 78.4 | 398.0 |
| **Pooling Types** | | | | | | | |
| Max-Pooling | 34.8 | 65.1 | 76.7 | 20.7 | 50.1 | 64.2 | 311.7 |
| Average-Pooling | 58.8 | 85.4 | 91.9 | 42.4 | 71.5 | 81.8 | 431.9 |
| Sum-Pooling | 58.4 | 85.1 | 92.1 | 41.3 | 70.5 | 80.7 | 428.1 |
| Softmax-Pooling | 54.7 | 83.0 | 91.3 | 40.3 | 70.0 | 81.0 | 420.5 |
| LSE-Pooling | 60.2 | 85.9 | 92.4 | 41.7 | 71.5 | 81.7 | **433.4** |

Quantitative Comparison between different coding setting.
Aligning sub-regions with query words yields the best results.




The impact of the codebook size.
Increasing the codebook size consistently improved the performance of CHAN, demonstrating its robustness.

# Experiments

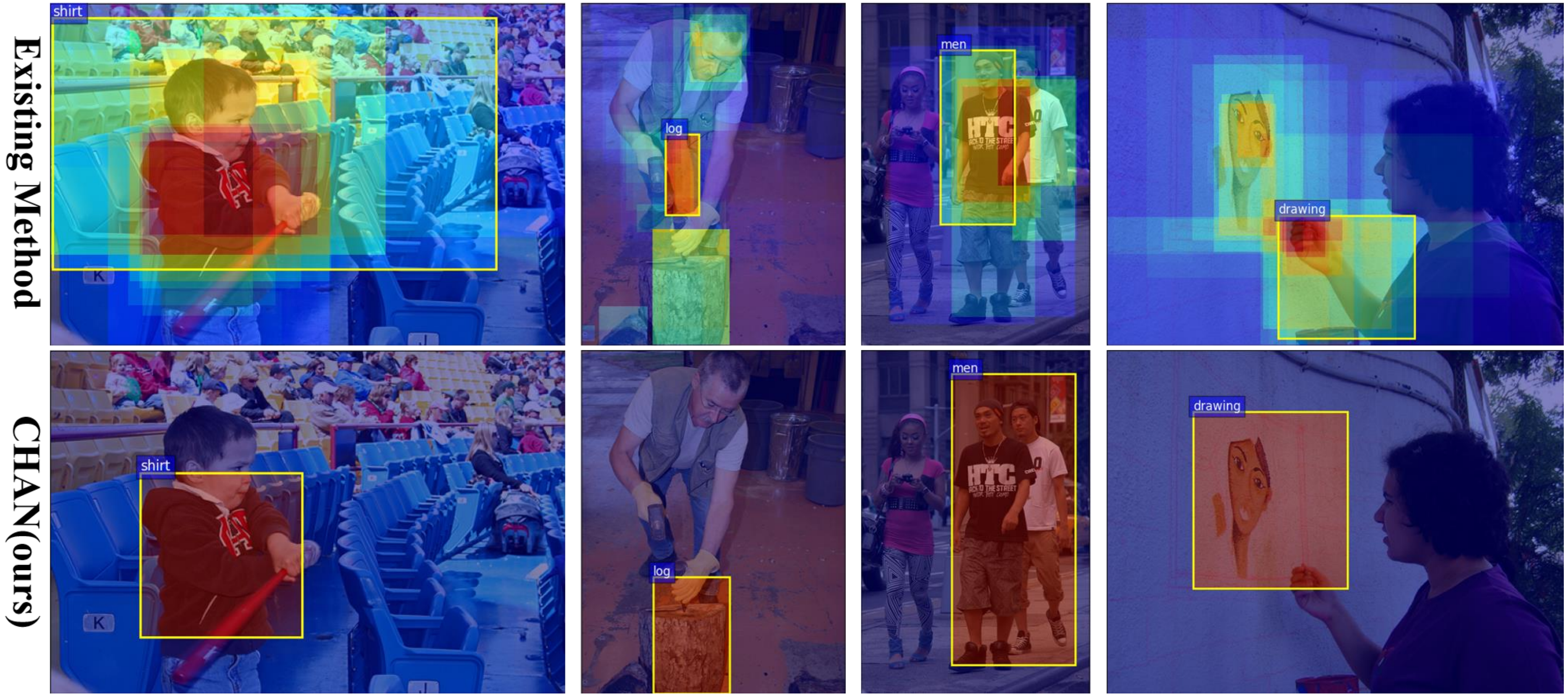


**Q1:** The boy wearing a black shirt and blue jeans is holding a red baseball bat.

**Q2:** A gloved hand holds what appears to be an oversize nail against a log.

**Q3:** Two men and a woman are walking down a city street.

**Q4:** A woman drawing a portrait on a white wall with trees in the background.




Visualization comparison between CHAN and existing method.
CHAN can better eliminate the meaningless alignments

# Thank you for Listening!

# References

[1] Pan Z, Wu F, Zhang B. Kernel triplet loss for image-text retrieval[J]. Computer Animation and Virtual Worlds, 2022, 33(3-4): e2093.
[2] Lee K H, Chen X, Hua G, et al. Stacked cross attention for image-text matching[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 201-216.