Tensor Factorization for Leveraging Cross-Modal Knowledge in Data-Constrained Infrared Object Detection
Most object detectors work well when provided with sufficient training data.

- Suffer overfitting due to over-parametrization in data scarce regime.
- RGB trained model does not generalize well to infrared/thermal due to significant domain shift.

Our task: Object detection in the data scarce infrared (IR) domain.

Given: Large amount of publicly available RGB training data.

Research questions:

- How to achieve generalizability for object detection from few labelled IR training samples?
- Can we leverage the abundance of annotated RGB data for object detection, in the IR domain?
Related Works

- Domain-Adaptive Pedestrian Detection in Thermal Images [1].


---

Motivation: Relatively Scarce IR Data

Challenges in acquiring IR data:

▪ Hardware cost and constraints (less ubiquitous than RGB cameras).
▪ Expensive and time-consuming data annotation process.
▪ Privacy concerns and export control regulation.

There exists common feature cues in both RGB and IR data.

▪ Exploit cross-modal cues at the model level.

Advantages of domain adaptation methods:

▪ Reduce data acquisition efforts.
▪ Reduce computational costs.
Contributions

- **TensorFact**: A novel tensor factorization method that can leverage both:
  - modality-specific cues.
  - cross-modal cues.

  for effective object detection in the IR data, where acquiring sufficient training data is a challenge.

- **TensorFact** outperforms the competing state-of-the-art object detector trained directly on data scarce target IR domain while retaining source RGB domain performance.
Technical Background

Convolution layer: $X^{S \times H \times W} \ast K^{T \times S \times D_2 \times D_1} = Y^{T \times H' \times W'}$

Input          Convolution Filter (Trainable Parameters)        Output

Tensor: n-D array
- 1-D array – Vector
- 2-D array – Matrix
- ≥3-D array – Tensor

Decomposed convolution filter:
- $[M]_{p,q} = \sum_{c=1}^{r}[A]_{p,c}[B]_{c,q}$
  - $p = 1, 2, ..., TS$
  - $q = 1, 2, ..., D_2 D_1$
- $[K]_{t,s,d_2,d_1} = [M]_{(t-1)S+s,(d_2-1)D_1+d_1}$
  - $t = 1, 2, ..., T$
  - $s = 1, 2, ..., S$
  - $d_2 = 1, 2, ..., D_2$
  - $d_1 = 1, 2, ..., D_1$
Proposed Method — TensorFact

**TensorFact:** Designed to tackle data scarcity in the IR data.

**For RGB:** Low-rank decomposed convolution filter.

**For IR:** Capacity augmentation

---

For Standard Convolution Filter:
- # trainable parameters \( (P) = TS D_2 D_1 \)

For RGB: \( A \) & \( B \) are SVD initialized
- # trainable parameters \( (P_{fac}) = r(TS + D_2 D_1) \)

In general, \( 0 < r \leq r_{max} \), \( r_{max} = \min(TS, D_2 D_1) \).

For varying \( r \) across network layers using a single variable - Introduce \( \alpha \) hyperparameter.
- \( r = \alpha r_{max}, \alpha \in (0,1) \)
- \( P_{fac} = \alpha r_{max}(TS + D_2 D_1) \)

For IR:
- # trainable parameters \( (\Delta P_{fac}) = \Delta r(TS + D_2 D_1) \)
  - \( \Delta r = \Delta \alpha r_{max} \)
2-Branch Architecture:

To promote learning complementary features, we propose the following loss term:

- \( \max(\|K(X) - \Delta K(X)\|_p), \ p = \{1,2\} \)

To increase the distance between feature maps.
Baseline: YOLOv7 [1]
- # Trainable Parameters: about 37 M.

Datasets:
- FLIR Aligned RGB [2]
  - Classes: Person, Bicycle, and Car.
- FLIR ADAS v1 IR [3]
  - Classes: Person, Bicycle, and Car.
  - Dataset Configuration:
    - Data constrained (Use only 1% of training data).

Evaluation Metric:
- Mean Average Precision (mAP) = $\frac{1}{n_c} \sum_{i=1}^{n_c} \bar{P}_i$
  - mAP 50
  - mAP 50-95

## Results

<table>
<thead>
<tr>
<th>Model</th>
<th># Parameters (M) ↓</th>
<th>Compression (%) ↑</th>
<th>mAP 50 (%) ↑</th>
<th>mAP 50-95 (%) ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLOv7</td>
<td>37.21</td>
<td>0</td>
<td>68.26</td>
<td>31.73</td>
</tr>
<tr>
<td>TensorFact ($\alpha = 0.9$)</td>
<td>35.40</td>
<td>4.85</td>
<td><strong>69.48</strong></td>
<td>31.62</td>
</tr>
<tr>
<td>TensorFact ($\alpha = 0.8$)</td>
<td><strong>33.59</strong></td>
<td>9.71</td>
<td>68.79</td>
<td>31.68</td>
</tr>
</tbody>
</table>

Results for FLIR Aligned RGB validation dataset

<table>
<thead>
<tr>
<th>Model</th>
<th># Parameters ↓</th>
<th>Compression (%) ↑</th>
<th>mAP 50 (%) ↑</th>
<th>mAP 50-95 (%) ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLOv7</td>
<td>37.21</td>
<td>0</td>
<td>58.49</td>
<td>28.07</td>
</tr>
<tr>
<td>TensorFact ($\alpha = 0.1$)</td>
<td><strong>1.86</strong></td>
<td><strong>95.01</strong></td>
<td>62.05</td>
<td>28.07</td>
</tr>
<tr>
<td>TensorFact ($\alpha = 0.2$)</td>
<td>3.66</td>
<td>90.16</td>
<td><strong>62.13</strong></td>
<td>27.94</td>
</tr>
</tbody>
</table>

Results for FLIR ADAS v1 IR validation dataset

<table>
<thead>
<tr>
<th>Regularization</th>
<th>mAP 50 (%) ↑</th>
<th>mAP 50-95 (%) ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>N/A</td>
<td>62.05</td>
<td>28.07</td>
</tr>
<tr>
<td>$L_1$</td>
<td><strong>62.34</strong></td>
<td><strong>28.23</strong></td>
</tr>
<tr>
<td>$L_2$</td>
<td>62.22</td>
<td>28.15</td>
</tr>
</tbody>
</table>

Results with explicit complementary regularization for $\alpha = 0.1$ on FLIR ADAS v1 IR validation dataset
Qualitative Results

YOLOv7 fails to detect small and distant objects, but TensorFact can detect them.
Summary:
We propose TensorFact—a method to architecturally promote learning of cross-modal cues.

- Improve generalization for modalities with scarce training data (as low as 62 samples).
- Require only a fraction of trainable parameters (5% of total parameters).
- Empirically validated the efficacy of our method for object detection.

Future Work:

- Explore attention between RGB and IR branches during forward pass to reduce false detection.
- Extend to other applications (e.g. segmentation).
Thank you!

Questions?