Appendix A
Gradient Derivation for the Embedding Network

In this section, we present $\frac{\partial \mathcal{L}_{\text{obj}}}{\partial \boldsymbol{W}_q}$, the gradient of $\mathcal{L}_{\text{obj}}$ with respect to $\boldsymbol{W}_q$, $\forall q \in [Q]$, for a stochastic gradient descent. To illustrate the gist, we consider the following simplistic setting. With a K-th order affinity tensor $\boldsymbol{\mathcal{T}}^{(K)} \in \mathbb{R}^{\overbrace{m \times m \cdots \times m}^{K}}$ as an example, for a possible iteration t, a given mini-batch sample $\boldsymbol{X}_{I_t}$, and a fixed weight matrix $\boldsymbol{W}_{\text{ort}}^{(t)}$, one can forward pass through TSC-Net to obtain $\boldsymbol{H}_{I_t}^{q}$, for $q \in [Q]$, with $\boldsymbol{Y}_{I_t}^{t}$, and construct a small part of the K-th order affinity tensor $\boldsymbol{\mathcal{T}}_{I_t}^{(K)} \in \mathbb{R}^{\overbrace{m_b \times m_b \cdots \times m_b}^{K}}$. Subsequently, using the chain rule, it follows that the gradient of $\mathcal{L}_{\text{obj}}$ with respect to $\boldsymbol{W}_Q^{(t)}$ is:

$$\frac{\partial \mathcal{L}_{\text{obj}}}{\partial \boldsymbol{W}_Q^{(t)}} = -(\boldsymbol{H}_{I_t}^{Q-1})^{\top} \cdot \boldsymbol{M}_{I_t} \cdot \boldsymbol{W}_{\text{ort}}^{(t)\top} \circ \text{sign}(g(\boldsymbol{H}_{I_t}^{Q-1}\boldsymbol{W}_Q^{(t)}))$$

, where $\circ$ denotes Hardmard product and sign is the element-wise sign function, *i.e.*, $\text{sign}(\boldsymbol{A}_{i,j}) = 1$ when $\boldsymbol{A}_{i,j} > 0$ and $\text{sign}(\boldsymbol{A}_{i,j}) = 0$ when $\boldsymbol{A}_{i,j} \leq 0$. The intermediate matrix $\boldsymbol{M}_{I_t} \in \mathbb{R}^{m_b \times c}$ means only taking corresponding part of $\boldsymbol{M} \in \mathbb{R}^{m \times c}$ on the basis of I_t, and $\boldsymbol{M}$ is affinity-tensor-dependent, where

- For second order affinity tensor $\boldsymbol{\mathcal{T}}^{(2)}$, $\boldsymbol{M} = 2\boldsymbol{\mathcal{T}}^{(2)}\boldsymbol{Y}$.
- For third order affinity tensor $\boldsymbol{\mathcal{T}}^{(3)}$, $\forall s \in c$, $\boldsymbol{M}_{:,s} = \boldsymbol{I}_m \boldsymbol{\mathcal{T}}_{(1)}^{(3)}(\boldsymbol{Y}_{:,s} \otimes \boldsymbol{Y}_{:,s}) + (\boldsymbol{I}_m \otimes \boldsymbol{Y}_{:,s} + \boldsymbol{Y}_{:,s} \otimes \boldsymbol{I}_m)^{\top}(\boldsymbol{\mathcal{T}}_{(1)}^{(3)})^{\top}\boldsymbol{Y}_{:,s}$, where $\boldsymbol{\mathcal{T}}_{(1)}^{(3)}$ denotes the Mode-1 unfolding of $\boldsymbol{\mathcal{T}}^{(3)}$ [44], and $\otimes$ denotes the Kronecker product.
- For fourth order affinity tensor $\boldsymbol{\mathcal{T}}^{(4)}$, $\forall s \in c$, $\boldsymbol{M}_{:,s} = \boldsymbol{I}_m \boldsymbol{\mathcal{T}}_{(1)}^{(4)}(\boldsymbol{Y}_{:,s} \otimes \boldsymbol{Y}_{:,s} \otimes \boldsymbol{Y}_{:,s}) + (\boldsymbol{I}_m \otimes \boldsymbol{Y}_{:,s} \otimes \boldsymbol{Y}_{:,s} + \boldsymbol{Y}_{:,s} \otimes (\boldsymbol{I}_m \otimes \boldsymbol{Y}_{:,s} + \boldsymbol{Y}_{:,s} \otimes \boldsymbol{I}_m))^{\top}(\boldsymbol{\mathcal{T}}_{(1)}^{(4)})^{\top}\boldsymbol{Y}_{:,s}$, where $\boldsymbol{\mathcal{T}}_{(1)}^{(4)}$ denotes the Mode-1 unfolding of $\boldsymbol{\mathcal{T}}^{(4)}$.

One can continue using the chain rule to obtain $\frac{\partial \mathcal{L}_{\text{obj}}}{\partial \boldsymbol{W}_q}$ for $q \in [Q-1]$.

Appendix B
Experiments on Noisy CMU-PIE to Demonstrate Noise-Robustness

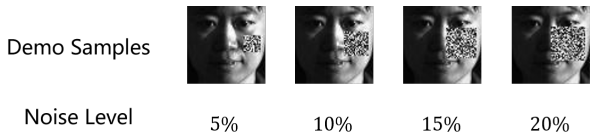

Fig. B1: Noise contamination demonstration on CMU-PIE dataset. The noise level increases from the left (5%) to the right (20%). The noise is generated by random Gaussian distribution with a mean of 0.5 and a variance of 0.25.

In practice, data usually suffer from noise contamination, which can be a leading factor that compromises the performance of most clustering methods. To test the noise-robustness of TSC-Net, we conducted a noise experiment on CMU-PIE data. CMU-PIE[2] comprises 1632 face images belonging to 68 individuals, where each image is represented by 4096 pixels. First, to simulate a real scenario, we added random Gaussian noise with mean 0.5 and variance 0.25 to each CMU-PIE image, where a demonstration is shown in Fig. B1. To be specific, we added the noise random Gaussian noise with varying levels ranging from 0% to 20%, as can be seen from left to right in Fig. B1, which mounts to yielding five datasets. Then, we applied the methods mentioned before to conduct experiments on the resultant datasets under the same computational protocols as stated in Section 4.1.

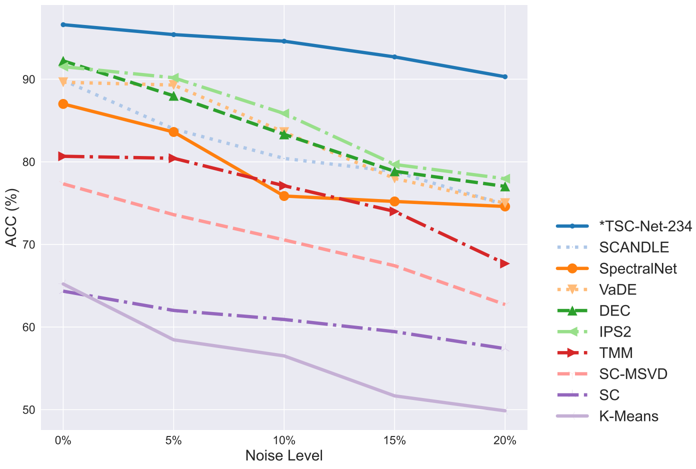

Fig. B2: ACC results of clustering methods on CMU-PIE dataset with increasing noise levels. The figure shows the ACC performance of each method as the noise level increases, where different methods are denoted by different lines.

In Fig. B2, we reported the ACC performance of TSC-Net as well as its competitors, as mentioned earlier, on noisy CMU-PIE with varying noise levels. From the figure, we underline the following observations:

- The curves corresponding to the proposed TSC-Net, denoted by the solid blue line with circles, are located at the top in Fig. B2, indicating its best overall performance. These results demonstrate that TSC-Net is capable of dealing with data under different levels of noise contamination.
- TSC-Net is more noise-robust than other methods. Although all methods face a performance drop with the increasing noise levels, as shown in Fig. B2, TSC-Net obtains a minor performance decline. For example, with the noise level from 0% to 20%, the ACC decrease of TSC-Net is 6.3%, whereas SpectralNet, which is the second-best stable method, receives an ACC decrease of 12.4%. This implies that TSC-Net is relatively more stable against noise contamination in comparison with other methods.

[2] http://www.cs.cmu.edu/afs/cs/project/PIE/MultiPie/Multi-Pie/Home.html.

- The representative TSC method, IPS2, receives a smaller ACC decrease than the classic deep clustering, DEC, with increasing noise levels. Notably, as the noise increases from 0% to 20%, the ACC decrease of IPS2 is 13.6%, while the one of DEC is 15.2%.

The superiority of the proposed TSC-Net with regard to noise-robustness arises from the characterization of multi-wise similarities. In detail, first, TSC-Net considers the multiple affinity tensor with various multi-wise similarities and integrates them to estimate a consensus embedding. The multiple affinity tensors integration enables us to alleviate the underlying noise contamination. In contrast, classic deep clustering methods, like DEC, consider the pairwise similarities to partition data, which are sensitive to noise even with a small level. Second, conducting the multiple affinity tensor integration in a one-stage way, *i.e.*, TSC-Net, enables a better noise-robustness than the two-stage method, IPS2. To see this, one can notice that the performance drop of TSC-Net is much smaller than that of IPS2 (6.3% vs. 13.6%).