We sincerely thank the reviewers for their valuable comments. We proofread and fixed the mentioned errors.

Broader Impact: Since our method improves subspace clustering, it advances learning from unannotated data. Improving the learning process and providing more accurate similarity matrices for unannotated data can positively impact accountability, transparency and explainability of AI methods. However, if not controlled, providing the opportunity to learn from big unannotated datasets could increase the concerns about violating the privacy of individuals.

Related Work: Thank you for the additional references. We will include and discuss them in the revised version.

Publishing codes: Upon the acceptance of our paper, we will publicly release the source codes.

Equation 9: While in MLRDSC (Eq. (9)), Q^T is the same as $\hat{Q}^T$, in our method, Q^T relates to $\hat{Q}^T$ by Eq. (4).

Replacing $\mathbf{Q^t_{Ens}}$ with $\hat{\mathbf{Q}}$ in Eq. (7) could also make an alternative consistency loss. However, using Q^t_{Ens} has the advantages of: 1) Ensembling multiple subspace membership predictions which results in a better estimate. 2) Providing smoother transitions for subspace membership predictions that could result in better learning.

Motivations behind augmentations for subspace clustering: Additional unlabeled data can only improve the representation learning in encoder and decoder, and not necessarily the self-expressiveness layer's parameters (which scales quadratically with additional data). Augmentations in our model regularize parameters in both encoder and decoder as well as the self-expressiveness layer. Intuitively, if two samples are similar, our method makes sure they remain similar even if they undergo transformations that do not change their label. Adding additional data does not provide such property.

Rationale Behind the Augmentation Search: Note that our augmentations search algorithm does not minimize Eq. (8). It looks for augmentation policies that yield the highest mean Silhouette coefficient. The rationale is similar to supervised models. Augmentations may initially increase the value of the loss function, but they lead to improved learning by regularizing the training.

Relation to metric learning and stochastic subspace clustering: Our model shares more similarities to stochastic subspace clustering (reference [17] in the submission) models (SSSC) than to metric learning methods. In SSSC, stochastic transformations are applied to the inputs of the subspace clustering algorithm. In addition, many subspace algorithms apply nonlinear functions to deal with nonlinearity (references [8,9,10,11] in the submission). We add the discussion in the final paper.

Silhouette coefficient: Silhouette coefficient measures how similar an object is to its own cluster compared to other clusters. As we assume in a clustering task we do not have any labels, an *external evaluation* such as Silhouette can be used as an estimation of clustering performance to avoid bad augmentations.

Experiments: We include the following experiments in the revision:

- **Runtime**: In Table 1, we report the runtime for the policy search algorithm on the COIL-20 dataset.
- **Silhouette Coefficient v.s. Ground-truth**: We tested our augmentation algorithm with accuracy as the Score on the COIL-20 dataset, and observed that for a set of 9 best augmentation policies it achieves the same clustering error rate of 1.79. Our found augmentations in Table 1 of the main paper are also among these best augmentation policies.
- **Search Baselines**: In Table 2, we compare the clustering error rate of training our method with the augmentations found by 1) our algorithm, 2) uniform sampling, and 3) coordinate descent. We test on the Extended Yale-B dataset.
- **Downstream tasks**: We use the latent space features from our network and from MLRDSC to perform a classification task. We randomly set 50% of the learned latent space features for ORL samples as test set and use the remaining samples in training an SVM model. Table 3 shows that features learned via our method are better for classification.
- **Training deeper networks**: We add two more layers to DSC network (denoted by DSC-5layer). Table 4 compares the clustering error rate of DSC-5layer with and without augmentations applied to the ORL dataset.

Table 1: Runtime.	Table 2: Error rates on Yale-B. Search baselines.			Table 3: Downstream task: Accuracy of SVM on ORL.		Table 4: Error rates on ORL. Deeper networks: DSC-5layer.	
COIL-20	Ours	Uniform	Coordinate Descent	Ours	MLRDSC	Without Aug.s	With Aug.s
261 mins	**0.82**	3.72	0.95	**95.5**	93	22.50	**13.25**

Details: We include the following details in the revision:

- **EMA decay**: We used the silhouette coefficient as an evaluation metric in cross-validation.
- **Statistical Significance**: In all the conducted experiments, we reported 5-fold averages.
- **Minibatchs**: Similar to other DSC methods, we input the whole dataset as a batch.
- **Policies with $< \ell_{max}$ sub-policies** were also considered as augmentation candidates.