We thank the reviewers for their thoughtful feedback and effort. It is a pleasure to receive such meaningful reviews.

Texture-less Patches (R1). Our model handles ambiguity by forming “soft" matches: in contrast to previous work [5], it considers a distribution over possible matches. Moreover, we can show that when a match is fundamentally ambiguous, the gradients of the loss are not detrimental because they “cancel": e.g., when a set of potential matches are identical (e.g. texture-less patches of the sky), the matching distribution will be perfectly uniform over this set, and the sum of gradients they contribute will be zero in the worst case. We will provide a proof in an updated version.

Partial Occlusions (R1, R2). The soft matching formulation is also robust to partial occlusion; by considering a distribution over matches for each node, it can form many-to-one and one-to-many matches, as well as matches at a region-level. Moreover, edge dropout (as well as spatial jittering) at training-time simulate similar challenges, providing further robustness, as can be seen in the improvement in J-recall in Table 1.

Object Moves out of Camera (R1, R2). Indeed, object disappearances do occur in less curated videos, such as those of our training set, Kinetics. The fact that our model learns a useful representation despite their presence suggests that it is robust to such cases. We hypothesize this is because total occlusions are infrequent and thus contribute only occasional label noise into the soft matching problem. Moreover, the use of sub-cycles (L148) also allows for learning from sub-sequences that may not be affected by total occlusions. Interestingly, we found that skip connections in time used to address occlusion in previous work [5] to not be beneficial. Incorporating additional temporal context in the right way is an important direction for future work.

Shortcut Solutions (R1, R2, R4). As the reviewers point out, learning from raw video involves the challenges mentioned above, such that it may be easier to learn a shortcut solution than model pixel appearance. As we mentioned in Footnote 1 (L145) and Appendix C (L9), we observed that naively using a single spatial feature map to obtain node embeddings at training time can lead to the network solving the matching objective without using visual appearance. This is evidenced by a solution obtaining very low training error (despite a large training dataset), even when asked to match high-resolution maps, while transferring poorly to downstream label propagation tasks. It can also be qualitatively observed in visualizations of feature maps, which we will add to the Appendix. Even after reducing boundary effects (which have been shown to be useful for encoding positional information [3]) by considering architectures without padding, as well as mitigating leakage of global information through batch normalization [2], we found that a shortcut could still be learned. Ultimately, we adopt a simple solution used in prior work [1, 4]: removing relative position cues altogether, by cropping the image to obtain patches. This works because predicting relative position of crops is challenging [1, 4]. This leads to meaningful modeling of visual appearance, as indicated in our evaluation experiments.

Feature Extraction (R1, R3). We extract separate patches at training time. At test-time, we can efficiently compute node embeddings by creating a single feature map and then flattening the spatial dimensions. We follow previous work [5]: our encoder makes a feature map 1/8 the the image resolution (e.g. 112×60 for 900×480px test images), and the label map is up-sampled with nearest-neighbor interpolation for evaluation.

Contrastive Learning with “Latent" Views (R1). By “latent" view, we mean that the corresponding positive view for each query is not known, as is the case in learning correspondence without labels; this is in contrast to typical contrastive learning settings, in which positive views are generated by hand-crafted data augmentation strategies. As scenes changes between frames, it is rare that corresponding nodes have exactly the same representation; it is invariance to spatio-temporal changes, rather than semantics, that we seek to learn. In our work, we leverage a path-level constraint (cycle-consistency) to guide the inference of latent views. We will clarify this in an updated version.

Image Classification Tasks (R3). Our main interest is a sense of similarity useful for finding space-time correspondence. While approaches like MoCo [2], seek invariance across instances of the same class, our representation distinctly should not. We view these two aims to play complementary roles in the spectrum of information needed to model instances and categories, in that we capture the natural data augmentation exhibited in dynamic scenes. More technically, unlike methods that rely on negatives drawn from other examples, our negatives come from within instance itself, so as to distinguish its parts from one another. That said, it is definitely interesting to consider how spatio-temporal invariance can give rise to semantic representations, and we will investigate image classification in an updated version.

Extending Related Work (R1, R3). We completely agree that connections to recent work on attention-based architectures for encoding sets and graphs, as well as the work of Dwibedi et al, are interesting and deserve discussion. If accepted, we look forward to using the extra page in the camera-ready to do so.

Clarification in Pseudocode (R3). We appreciate that reviewers took the time to read our pseudocode, and hope it was of help. Indeed, the output dimension of that operation is `B x (T-1) x P x P`. We will clarify this.

Training Hyper-parameters (R1, R3). While our hyper-parameters (Appendix E) follow prior work [2, 4] due to computational constraints, we agree that leveraging more recent hyper-parameters might result in further improvements.

[1] C Doersch et al. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015.
[2] K He et al. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.
[3] M Islam et al. How much position information do convolutional neural networks encode? *ICLR*, 2020.
[4] A Oord et al. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.
[5] X Wang et al. Learning correspondence from the cycle-consistency of time. In *CVPR*, 2019.