1 Key Ideas

1. Use unsupervised learning to detect keypoints, which are interpreted as concise object representations.
2. This is done by enforcing a keypoint bottleneck when learnt image features are transported from a source to a target frames
3. Benefit: the discovered keypoints track objects more consistently across long-time horizons.
4. Benefit wrt RL: using the keypoints and corresponding image features enable more sample efficient RL.
5. Benefit wrt RL: exploration by controlling keypoints locations enables deeper exploration, reaching states not reached by random exploration.

2 Unsupervised Detection of keypoint: Feature Transport

1. The main idea is this: given a src img and tgt img, to accurately construct the tgt img, the network needs to erases the portion in the src image which will not appear in the tgt img, and inpaint the features that do not exist in the src img, but will exist in the tgt img.
2. The main assumption is that the two imgs only differ in obj's pose, geometry or appearance.
3. To do this, two types of networks are trained: a generic feature extractor and a keypoint extractor.
4. Given src img $\boldsymbol{x}_t$ and tgt img $\boldsymbol{x}_{t'}$, we have generic feature map $\Phi(\boldsymbol{x}_t)$, $\Phi(\boldsymbol{x}_{t'})$ and extracted keypoints $\Psi(\boldsymbol{x}_t)$, $\Psi(\boldsymbol{x}_{t'})$.
5. We then generate Gaussian heatmap by constructing isotropic fixed-variance Gaussians around each of the keypoints.
6. And then, a composite feature map is generated by: setting the features around the keypoints extracted from the src img to 0, replacing the features in the srg img around the keypoints extracted from the tgt img with the features from the same locations in the tgt img.
7. Thus, the keypoint extraction network needs to correctly identify positions which differ between the src and tgt img. Because of the assumptions that the 2 imgs only differ in quantities relevant to the objs, the keypoint extractor is thus forced to extract keypoints corresponding to the objs.
8. A point which they stress is that the approach, in contrast to previous approach which stacks the keypoint heatmas, enforce "explicit spatial transport for stronger correlation with img locations leading to more robust long-term tracking".
9. I think what they mean is that by compositing the features corresponding to the extracted keypoints fom the tgt img to the same img, the relative geometrical relationship between the extracted keypoints are forced to correlate more with the locations of objs in the original tgt img.

3 Using keypoints for exploration in RL

1. They construct new action space, where each action corresponds to controlling the movement of one keypoint in one spatial direction.
2. Concretely, they train multiple Q functions, where each function corresponds to the change in the position of one keypoint in one direction.
3. During exploration, they pick a single Q functions and commit to this Q for a fixed number of timestep. This is a so-called option.
4. They also argue for picking the most controllable keypoint, by measuring the difference between the predicted maximum change in location and predicted minimum change in location.

4 Questions

1. One thing this paper does not consider is the interactions between the keypoints. For example, instead of selecting the most controllable policy, we can imagine selecting the policy that affects the most change on the environment, as measured by the changes in the position of all the keypoints.

5 Interesting tibits

1. They design an interesting metrics to measure consistent detection of key points across long-time horizon.
2. They mention that applying their approach to all Atari games will require training the unsupervised learning model inside the RL loop because data from a random policy is insufficient for games where new objects or screens can appear.