Improving Multi-Task Reinforcement Learning through Disentangled Representation Learning

Pranay Pasula
University of California, Berkeley
pasula@berkeley.edu

Abstract: When humans learn to perform a task, they tend to also improve their skills on related tasks, even without explicitly practicing these other tasks. In reinforcement learning (RL), the multi-task setting aims to leverage similarities across tasks to help agents more quickly learn multiple tasks simultaneously. However, multitask RL has a number of key issues, such as negative interference, that make it difficult to implement in practice. We propose an approach that makes use of *disentangled feature learning* to alleviate these issues and find effective multi-task policies in a high-dimensional raw-pixel observation space. We show that this approach can be superior to other multi-task RL techniques with little additional cost. Finally, we investigate disentanglement itself by capturing, adjusting, and reconstructing latent representations that have been learned from Atari images and gain insight into their underlying meaning.

Keywords: Multi Task Reinforcement Learning, Disentanglement, Generalization

1 Introduction

The ability to learn from diverse experiences in order to adapt to novel situations is generally accepted as a trademark of intelligence. Reinforcement learning (RL) offers a framework that enables machines to do this through approaches that resemble some ways that humans learn. RL has been effective in a number of domains, such as games [1] and robotics [2], but compared to other artificial intelligence fields, such as computer vision and natural language processing (NLP), it's impact has been limited.

Some key issues in artificial intelligence are data scarcity, large time and sample complexities, and poor generalization. Computer vision and NLP have alleviated these issues through transfer learning, for example by pretraining classification models on ImageNet [3] or by using word embeddings that were pretrained on large text corpuses [4]. However these issues remain largely unresolved in RL, in which tasks are generally less abundant, more specific, and more difficult to learn than images or text are. Furthermore, by learning certain tasks, an agent often slows, or even reverses, its progress on learning other tasks, a phenomenon known as negative interference [5].

We propose a novel multi-task reinforcement learning (MTRL) approach to address all of the issues listed above. Our approach makes use of recent advances in representation learning to summarize high-dimensional and computationally-expensive observation spaces into small sets of learned latent factors. The small size of these latent layers bottlenecks the amount of information that can pass through [6, 7], reducing the degree to which a learned multi-task agent overfits to any particular task while retaining much of the explanatory power of the original observation spaces. Furthermore, we induce *independence* and *interpretability*, together referred to as *disentanglement*, in the learned latent factors by penalizing the model for having latent factors that deviate from an isotropic prior. We gain intuition into latent representations of complex observation spaces by viewing images generated by learned latent factors, both by using the exact latent encodings from real images and by varying the activations of these once exact latent encodings to investigate the semantics of each learned latent factor.

We find that our approach can help alleviate the impacts of data scarcity, large time and sample complexities, poor generalization, and negative interference in MTRL on high-dimensional raw-pixel observation spaces.

2 Preliminaries

Problem formulation: We consider the standard RL problem formulation extended to a multi-task setting defined by $(\mathcal{S}_i,\ \mathcal{A}_i,\ \mathcal{T}_i(s_{i,t+1}|s_{i,t},a_{i,t}),\ p_i(s_{i,0}),\ r_i(s_{i,t},a_{i,t})$ for $i \in N$, where for each task i in N number of tasks, $\mathcal{S}_i$ is the state space, $\mathcal{A}_i$ is the action space, $\mathcal{T}_i$ is the transition function, $p_i(s_0)$ is the starting state distribution, and r_i is the reward function for that task.

β-**VAE formulation**: We assume that a small subset of generative factors $\{\mathbf{z}\}$ explains approximately all of the variance of any observation $\mathbf{x}$ that our agent can see. By using a β-VAE [8], we aim to learn $\{\mathbf{z}\}$. Therefore, a fitting objective is to maximize the expected log-likelihood of observed data $\mathbf{x}$ conditioned on the latent generative factors $\mathbf{z}$ over the distribution of $\mathbf{z}$. However instead of using the distribution of latent factors $\mathbf{z}$, we instead infer the posterior by introducting a distribution $q_\phi(\mathbf{z}|\mathbf{x})$ and taking its expectation over $\mathbf{x}$. This is referred to as the *reconstruction loss* term To encourage disentanglement among the latent factors, we penalize $q_\phi(\mathbf{z}|\mathbf{x})$ for deviating from a multivariate isotropic Gaussian prior. The resulting objective function is

$$\mathcal{L} = E_{q_\phi(\mathbf{z}|\mathbf{x})}[\log(p_\theta(\mathbf{x}|\mathbf{z}))] - \beta D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})||p(z)),$$

where β controls the degree to which the objective encourages latent factors to match the prior. In other words, higher values of β require posterior distributions to uniquely explain more of the reconstructed output in order to deviate from the prior distribution.

3 Improving Multi-Task Learning through Disentanglement

Improving Generalization through Disentanglement

Single-Task Stage

Our goal is to learn a tractable multi-task policy that can perform multiple tasks well simultaneously. We evaluate this multi-task policy in the domain of Atari games by comparing the performance of learned multi-task policies to the performance of a expert individual-task policies. Before learning expert policies, we use the approach in [9] to train an expert Deep Q-Network (DQN) on each task, except that our approach uses a piecewise linear exploration schedule instead of an ϵ-greedy approach.

Parisotto et al. [10] introduced Actor-Mimic, an algorithm that distilled individual task policies into multi-task policies, but this approach trained and evaluated agents on only the original observation spaces. To address the issues of overfitting and negative interference, before transforming the expert DQNs into expert policies, we train variational autoencoders (VAE) [11] to encode and reconstruct observations seen by expert DQNs in a fully unsupervised manner. By restricting the latent layer size, we restrict the amount of information that can flow through the autoencoder, known as the ”information bottlenecking” [6, 7]. This encourages the autoencoder to learn more general representations than it otherwise would have.

The VAEs have the following architecture: We use the architecture of the convolutional part of the DQNs as the architecture for the encoder. For the remainder of the VAEs, we have an intermediate FC layer between the encoder and the latent layer, a latent layer, and the reverse of the encoder as the decoder. Since DQNs are generally exposed to more of the state space as their performance improves, using observations seen by expert DQNs over non-expert DQNs encourages the VAEs to learn more general representations.

To further increase the generality captured by VAEs, we regularize the representations by penalizing the VAE the more that its latent posteriors deviate from a particular prior distribution. In our case, we specify an isotropic multivariate Gaussian prior, which transforms our VAE into the β-VAE introduced by Higgins et al. [8]. We vary the latent layer size over {16, 32, 64} and β over {1, 5, 25, 50}.

We then replace the convolutional part of each expert DQN with the encoder section of the corresponding learned VAE. We assume that the VAEs have indeed learned useful, general representations of the observations seen by the expert DQNs, so we freeze the weights of these convolutional sections while leaving FC layers free to change.

After supplanting the convolutional part of the DQNs, we create an expert stochastic policy for each task by taking the softmax over actions of each expert DQN. Equivalently, the resulting expert policy is the Boltzmann distribution (with temperature parameter equal to 1) of the corresponding expert DQN. The distributions of Q-values for different games are often very different, so a crucial benefit of this approach is that it maps all Q-values to the [0, 1] interval. Parisotto et al. [10] found that without this intermediate step, distilled multitask policies failed to learn useful behaviors. We found that even while leveraging disentanglement, the multitask still performed poorly when trained on Q-values directly.

Multi-Task Stage

We distill the expert policies into a single multi-task policy by minimizing the cross entropy loss between all of the expert policies and the multi-task policy. Here we have a choice to sample from the expert policies, the multi-task policy, or both to generate roll-outs for training. Parisotto et al. [10] found that their multi-task policy learned better by sampling from the multi-task policy, so for the sake of time, we did the same. However since our approach is different from Actor-Mimic, a different sampling strategy may lead to better results.

Overall Approach

To simplify the description of our proposed approach, we enumerate the high-level steps below.

1. For each source task, train an expert DQN on that task.
2. For each source task, use observations from the corresponding expert DQN to train a β-VAE with some specified latent layer size z and channel information constraint parameter β.
3. For each source task, replace the convolutional section of the expert DQN with the encoder of the corresponding β-VAE, and freeze the weights of this section.
4. For each expert DQN, add a fully-connected (FC) layer between the existing FC layer and the action output layer. We used an FC layer with 256 nodes, which is half the number of nodes in the pre-existing FC layer.
5. For each source task, train the new DQN until it becomes an expert DQN.
6. For each source task, create an expert policy by taking the softmax of the Q-values over the corresponding task action space.
7. Initialize the multi-task policy, which has the same architecture as the original expert DQNs (only one FC layer with 512 nodes).
8. Train the multi-task policy by minimizing the sum of cross entropy losses between the expert policies and the current version of the multi-task policy.

The Importance of Independence

To gain insight into the usefulness of independent, or even disentangled, latent factors, we set $\beta = 0$ and varied $|z|$ over $\{32, 64\}$ and ran the algorithm above on the four Atari games in our training set, *Beam Rider*, *Demon Attack*, *Phoenix*, and *Space Invaders*. In other words, we removed the regularizer in the VAE objective, leaving just the reconstruction loss term, converting the β-VAEs into vanilla autoencoders.

Reducing Negative Interference through Disentanglement

We induce negative transfer by training on four qualitatively similar games: *Beam Rider*, *Demon Attack*, *Phoenix*, and *Space Invaders*. Frames captured during gameplay for each of these games are shown in Figure [1] and indicate visual similarity. Specifically, all games have enemies that are similar in size and a black background comprises almost the entire observation. In all of these games, the player controls a space or aerial vehicle that they use to move left or right while shooting or dodging enemies.

We induce negative interference in the multi-task learning process by replacing *Space Invaders* with *River Raid*, a game with mechanics similar to the three remaining games but with very different visuals. Figure [1] illustrates this visual dissimilarity. We learn a multi-task policy using all of the steps in our proposed approach except for those involving VAEs. In other words, we directly use the first expert DQN we train to find a corresponding expert policy instead of using it to first train a VAE. As a result, we don't explicitly induce an information bottleneck, and therefore the representations learned by the DQN are prone to overfit to the single task being learned.

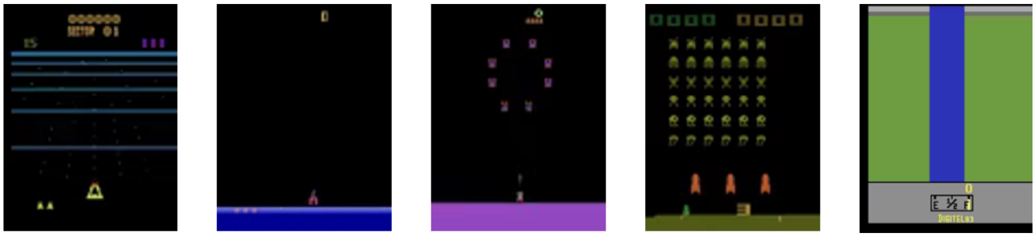


Figure 1: From left to right: *Beam Rider*, *Demon Attack*, *Phoenix*, *Space Invaders*, *River Raid*.

Since our proposed approach regularizes the factors used by the expert DQNs, we expect it to reduce negative interference resulting from visual dissimilarities across source tasks. We employ multiple renditions of our proposed approach with a latent layer size of 64 and with β varied over 1, 5, 25, 50.

Disentangling Disentanglement through Disentanglement

To the best of our knowledge, no prior work has specifically investigated disentanglement, or even information bottlenecks, in neural network representations learned from raw-pixel observations of Atari games. This is surprising because the Arcade Learning Environment is one of the most popular testbeds and benchmarks for developing and evaluating reinforcement learning algorithms. To gain insight into disentanglement and information bottlenecks, we train β-VAEs with various latent layer sizes and β values on observations from the Atari games discussed above.

Results

Improving Generalization through Disentanglement

To first gain intuition into how varying β affects a game in our training set, we view the performance of DQNs that have had their convolutional sections replaced with encoder sections of β-VAEs on the Atari game *Space Invaders*. The latent layer size was fixed at 64, and the values of β were varied over 1, 5, 25, 50. Figure [2] depicts the average returns of the DQNs over a training period of 4 million timesteps.

In the sections that follow, we evaluate our approach with various latent layer sizes $|z|$ and channel information constraint parameter β on multiple Atari games. We show that our approach can increase the generalization and robustness of a learned multi-task agent over another efficient approach to learning multi-task policies, Actor-Mimic.

The Importance of Independence

Results of our approach are shown in Figure [3], and the performance of single-task DQNs and of multi-task agents trained through Actor-Mimic are included for comparison.

We find that the multi-task agents from both Actor-Mimic and from our approach using $\beta = 0$ learn more quickly than their single-task counterparts. This is likely because the training tasks share characteristics in such a way that an agent that improves on one task simultaneously improves on the

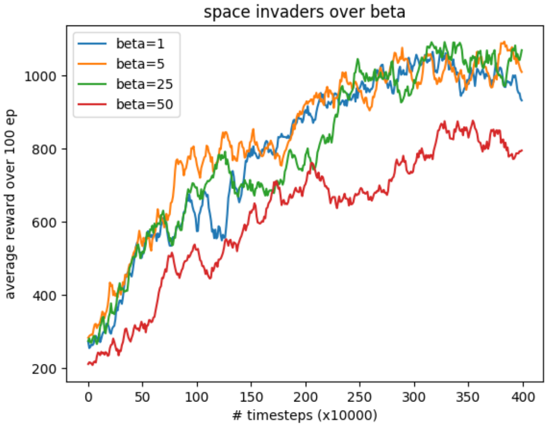


Figure 2: The performance of DQNs that have had their convolutional sections replaced with encoders of β-VAEs on *Space Invaders.*

other tasks even without training on the latter. However this is often not the case. A training task can be so different from others that an agent learning to play the first actually worsens its performance on the latter. We investigate this in the next section.

Reducing Negative Interference through Disentanglement

Originally, we trained a multi-task policy on games that all had similar visuals and gameplay. Since the algorithms under consideration involve agents that learn entirely from raw-pixel observations, we predict that different enough visual characteristics across source tasks will hinder an agents ability to learn all of them simultaneously. To induce negative interference into the multi-task learning process, we replaced *Space Invaders* with *River Raid*, which has similar gameplay to but very different visuals from *Beam Rider*, *Demon Attack*, and *Phoenix*, the other three games in the training task set as seen by Figure [1].

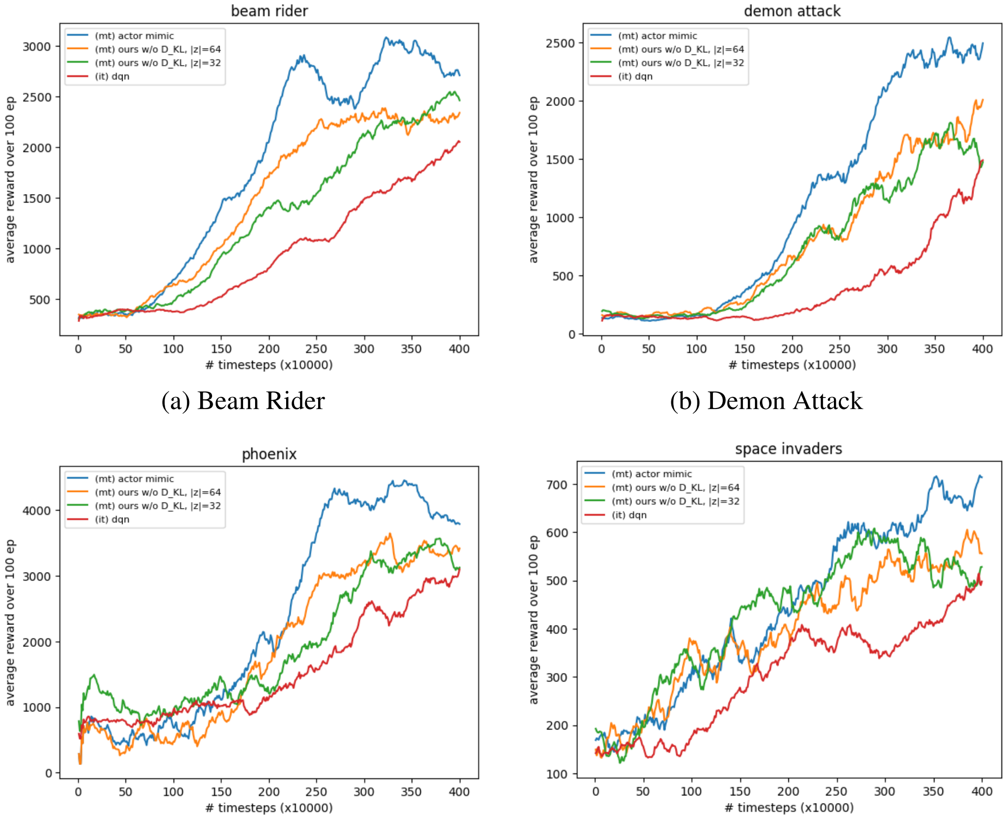
Since Actor-Mimic also uses policy distillation but does not attempt to learn latent representations, let alone disentangled ones, we use it as a baseline to assess how our proposed approach addresses negative interference in multi-task learning from raw-pixel observations.

The results shown in Figure [4] indicate that our approach can alleviate negative interference in multi-task learning.

Disentangling Disentanglement through Disentanglement

We trained β-VAEs on 200,000 observations each, which were obtained by running expert DQNs. Each β-VAE was trained for 5 epochs using a batch size of 32 and the RMSprop optimizer [**?**]. We randomly chose an observation, passed it through the β-VAEs, and extracted the latent activations that resulted. We then varied one latent factor while holding all others constant. Then we fed all the latent factors into the decoders of learned β-VAEs to obtain the reconstructed observations that correspond to these latent representations. To the best of our knowledge, this is the first work that explores disentanglement on the Atari domain.

We varied $|z|$ over $\{16, 32, 64\}$ and β over 1, 5, 25, 50, 100 and display the results below. Figure [5], Figure [6], and Figure [7] depict reconstructions from latent layers with sizes $|z| = 64$, $|z| = 32$, and $|z| = 16$ respectively.



(a) Beam Rider

(b) Demon Attack

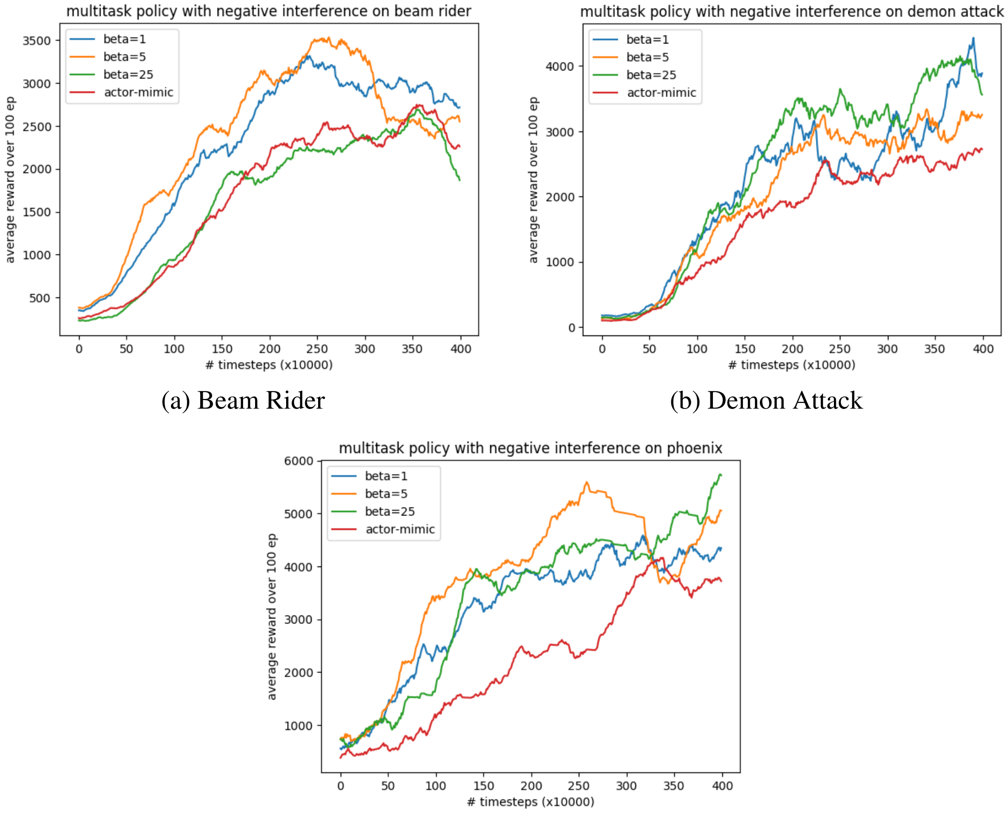
(c) Phoenix

(d) Space Invaders

Figure 3: Peformance of agents on Atari games trained by using our approach with latent layer sizes $|z|$ over $\{32, 64\}$ and β fixed at 0 compared to multi-task (mt) agents trained by using Actor-Mimic and individual-task (it) DQNs.

4 Conclusion

We identified issues in multi-task reinforcement learning and proposed a tractable algorithm that alleviates these issues in high-dimensional spaces. We evaluated DQNs that had their convolutional sections replaced with encoders from learned β-VAE and showed how different values of β affected performance on *Space Invaders*. We evaluated the importance of independent, or even disentangled, latent factors using our algorithm by pretraining β-VAEs over multiple values for β, including $\beta = 0$, which reduced the β-VAE into a vanilla autoencoder. We intentionally hindered MTRL by replacing a useful game in the training set with a harmful one and showed that our approach was superior to Actor-Mimic in reducing negative interference. However we also showed that without encouraging disentanglement (i.e. setting $\beta = 0$), Actor-Mimic actually outperforms our algorithm. We investigated the effects of information bottlenecking and disentanglement in the Atari domain over several latent layer sizes (i.e. degrees of information bottlenecking) and several values of β (i.e. degrees of disentanglement) and find a number of striking results.

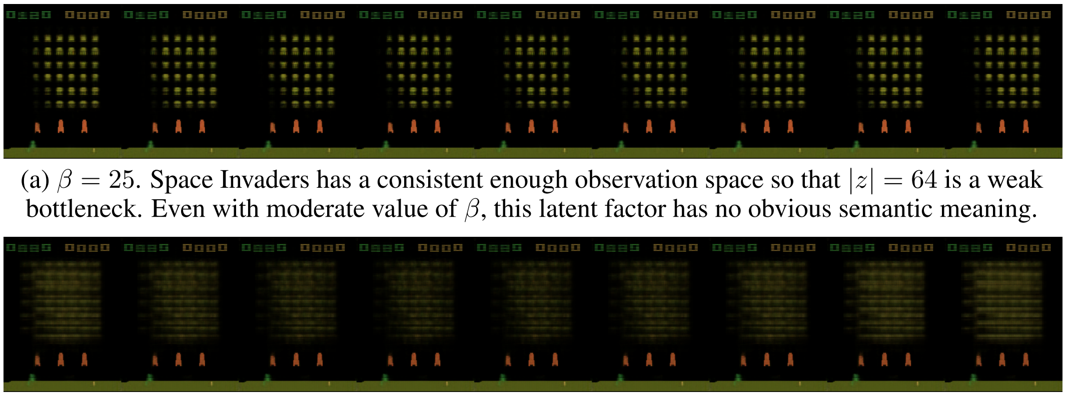


(a) Beam Rider

(b) Demon Attack

(c) Phoenix

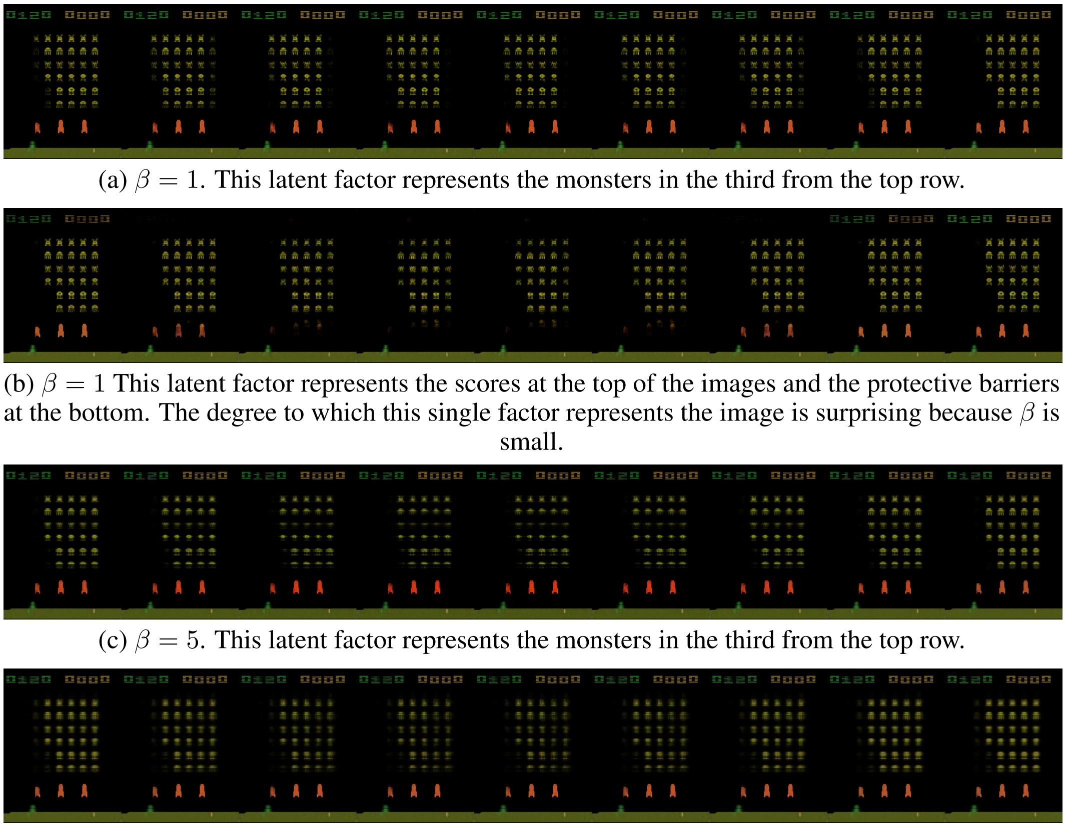
Figure 4: Peformance of multi-task agents on *Beam Rider*, *Demon Attack*, and *Phoenix* trained by using our approach with latent layer size $|z| = 64$ and β over $\{1, 5, 25\}$ compared to multi-task agents trained by using Actor-Mimic. To note, we used the double DQN algorithm here whereas the results shown in Figure [3] are from runs that did not use a target network.



(a) $\beta = 25$. Space Invaders has a consistent enough observation space so that $|z| = 64$ is a weak bottleneck. Even with moderate value of β, this latent factor has no obvious semantic meaning.

(b) $\beta = 100$ However, with a large enough β, the D_{KL} term in the objective function dominates, forcing the posterior distribution of this latent to match the isotropic Gaussian prior.

Figure 5: Some reconstructions from latent layers with size $|z| = 64$.



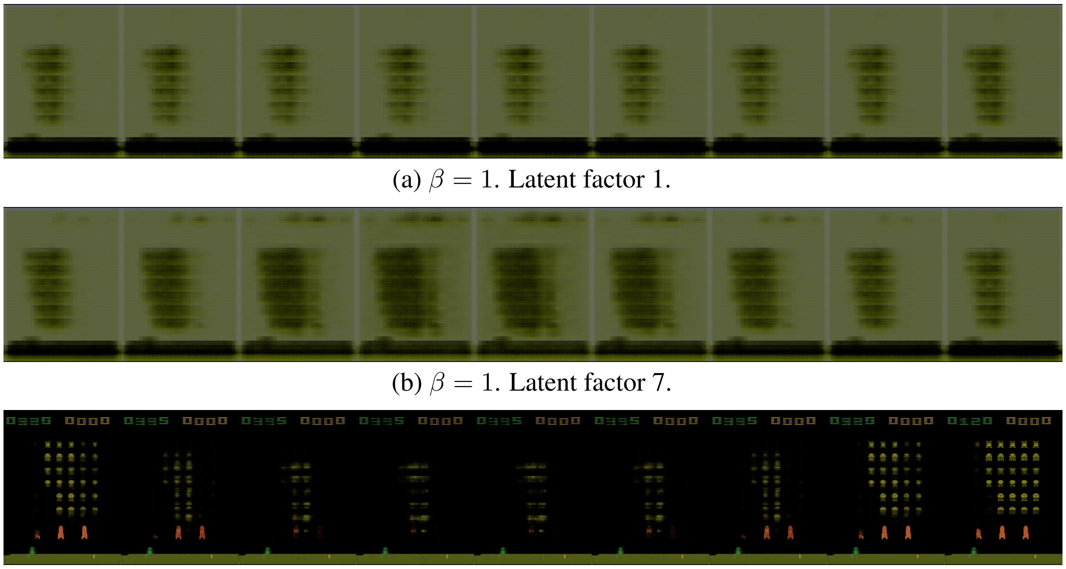
(a) $\beta = 1$. This latent factor represents the monsters in the third from the top row.

(b) $\beta = 1$ This latent factor represents the scores at the top of the images and the protective barriers at the bottom. The degree to which this single factor represents the image is surprising because β is small.

(c) $\beta = 5$. This latent factor represents the monsters in the third from the top row.

(d) $\beta = 25$ This latent factor represents the monsters in the two rightmost columns. To a lesser degree it also represents some rows of monsters.

Figure 6: Some reconstructions from latent layers with size $|z| = 32$.



(a) $\beta = 1$. Latent factor 1.

(b) $\beta = 1$. Latent factor 7.

(c) $\beta = 5$. Latent factor 2.

Figure 7: Some reconstructions from latent layers with size $|z| = 16$. (a) and (b) depict reconstructions from a β-VAE that collapsed to a useless local optimum. (c) show reconstructions after increasing β from 1 to 5, a small increase but enough to prevent the type of collapse seen in (a) and (b). Since $|z|$ is small, disentangled latent factors generally affect more of the image than they did for $|z| = 32$ or $|z| = 64$ as seen in Figure [7] and Figure [6] respectively.

Acknowledgments

Thanks to Pieter Abbeel for useful insights and continued interest in this work. Thanks to Sergey Levine for valuable feedback on previous iterations of our eventual approach.

References

[1] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller. Playing atari with deep reinforcement learning, 2013.

[2] J. Kober, J. A. Bagnell, and J. Peters. Reinforcement learning in robotics: A survey. *Int. J. Rob. Res.*, 32(11):1238–1274, Sept. 2013. ISSN 0278-3649. doi:10.1177/0278364913495721. URL `http://dx.doi.org/10.1177/0278364913495721`.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

[5] M. Mccloskey and N. J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *The Psychology of Learning and Motivation*, 24:104–169, 1989.

[6] N. Tishby, F. C. Pereira, and W. Bialek. The information bottleneck method, 2000.

[7] N. Tishby and N. Zaslavsky. Deep learning and the information bottleneck principle, 2015.

[8] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. M. Botvinick, S. Mohamed, and A. Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

[9] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, Feb. 2015. ISSN 00280836. URL `http://dx.doi.org/10.1038/nature14236`.

[10] E. Parisotto, J. L. Ba, and R. Salakhutdinov. Actor-mimic: Deep multitask and transfer reinforcement learning, 2015.

[11] D. P. Kingma and M. Welling. Auto-encoding variational bayes, 2013.