Spatially Invariant Attend, Infer, Repeat

Eric Crawford
Mila, McGill University
Montreal, QC
eric.crawford@cs.mcgill.ca

Joelle Pineau
Facebook AI Research,
Mila, McGill University
Montreal, QC
jpineau@cs.mcgill.ca

Abstract

The physical world can be naturally sub-divided into discrete objects. Consequently, in the pursuit of constructing ever more intelligent agents, devising methods for reasoning and learning about objects should be regarded as an important goal. Indeed, recent machine learning literature is replete with examples of the benefits of object-like representations when modeling the physical world (e.g. Chang et al. (2016)). However, in order to reason in terms of objects, agents need a way of discovering and detecting objects from visual input - a task which we call *unsupervised object detection*. This task has received significantly less attention in the literature than its supervised counterpart, especially in the case of large images containing many objects. In the current work, we develop a neural network architecture that effectively addresses this large-image, many-object setting. Through a series of experiments, we demonstrate a number of features of our architecture: that, unlike competing approaches, it is able to discover and detect objects in large, many-object scenes and that it has a significant ability to generalize to images that are larger and contain more objects than images encountered during training.

We introduce **Sp**atially Invariant **A**ttend, **I**nfer, **R**epeat (SPAIR), an architecture for unsupervised object detection that combines the unsupervised trainability of Attend, Infer, Repeat (AIR) (Eslami et al., 2016) with the spatial invariance properties of recent work in supervised object detection, particularly YOLO (Redmon et al., 2016). SPAIR is a Variational Autoencoder (VAE) (Kingma and Welling, 2013) with a highly structured, object-like latent representation z, a convolutional (and thus spatially invariant), object-detecting encoder network $q_\phi(z|x)$, and a decoder network $p_\theta(x|z)$ that "renders" detected objects into a reconstructed image. We now describe each of these components.

Object-like Latent Representation. We first describe the representation scheme used by SPAIR for describing objects in its latent layer. Given an image with shape $(H_{img}, W_{img}, 3)$, it will be useful to spatially divide the image into an (H, W) grid of cells, where $H = \lceil H_{img}/c_h \rceil$, $W = \lceil W_{img}/c_w \rceil$, and c_h/c_w are fixed integers giving the cell height/width in pixels. We employ a representation that allows a single object per cell (the extension to multiple objects per cell is straightforward).

For a cell with indices (i, j), $i \in \{0, \dots, H-1\}$, $j \in \{0, \dots, W-1\}$, the corresponding object is described by the following variables:

$$z^{ij}_{\text{what}} \in \mathbb{R}^A \qquad z^{ij}_{\text{depth}} \in \mathbb{R} \qquad z^{ij}_{\text{pres}} \in \{0, 1\} \qquad z^{ij}_{\text{where}} \in \mathbb{R}^4$$

z^{ij}_{what} is a vector with dimension A that stores appearance information for the object. z^{ij}_{depth} specifies the relative depth of the object; in the output image, objects with lower depth appear on top of objects with higher depth. z^{ij}_{pres} is a binary variable specifying whether the object exists and should be rendered to the output image. z^{ij}_{where} decomposes as $z^{ij}_{\text{where}} = (z^{ij}_y, z^{ij}_x, z^{ij}_h, z^{ij}_w)$, where z^{ij}_h and z^{ij}_w give the size of the object, and z^{ij}_y and z^{ij}_x give the position of the object relative to the current cell.

Modeling the Physical World workshop at NeurIPS 2018, Montréal, Canada.

Prior Distribution on Objects. A crucial component of a VAE is the prior distribution $p(z)$ over the latent variables. For all real-valued variables, we assume independent Normal distributions; the moments of these distributions are model hyperparameters. For the binary random variables z^{ij}_{pres}, we design a prior that puts pressure on the network to reconstruct the image using as few objects as possible (i.e. as few $z^{ij}_{\text{pres}} = 1$ as possible), similar to the prior used by AIR. This pressure is necessary for the network to extract quality object-like representations; without it, the network is free to set all $z^{ij}_{\text{pres}} = 1$, and use multiple latent objects to explain each object in the image.

Encoder Network. Our goal is to design an encoder network $q_\phi(z|x)$ with spatially invariant properties. To this end, a convolutional neural network $e^{conv}_\phi(x)$ is first used to map from the input image x to a feature volume with spatial dimensions (H, W). Next, this volume is processed sequentially cell-by-cell to produce objects, starting from the top left and proceeding row-by-row toward the bottom right.

Processing a cell runs as follows. First, a multi-layer perceptron e^{lat}_ϕ produces parameters $(\mu^{ij}_{\text{where}}, \sigma^{ij}_{\text{where}})$, $(\mu^{ij}_{\text{depth}}, \sigma^{ij}_{\text{depth}})$ and β^{ij}_{pres} for distributions over z^{ij}_{where}, z^{ij}_{depth} and z^{ij}_{pres}, respectively. As input, e^{lat}_ϕ accepts the feature vector (output of e^{conv}) at the current cell as well as sampled objects at nearby cells that have already been processed. e^{lat} can thus be thought of as encompassing a set of "lateral" connections which facilitate conditioning between nearby objects.

Next, values are sampled from the distributions. Concretely:

$$z^{ij}_{\text{where}} \sim N(\mu^{ij}_{\text{where}}, \sigma^{ij}_{\text{where}}) \qquad z^{ij}_{\text{depth}} \sim N(\mu^{ij}_{\text{depth}}, \sigma^{ij}_{\text{depth}}) \qquad z^{ij}_{\text{pres}} \sim \text{Bernoulli}(\beta^{ij}_{\text{pres}})$$

The sampled value of z^{ij}_{where} is then used along with a spatial transformer T (Jaderberg et al., 2015) to extract a glimpse from the image. This glimpse is processed by an *object encoder network* e^{obj}_ϕ to yield parameters for a distribution over z^{ij}_{what}, which is subsequently sampled:

$$\mu^{ij}_{\text{what}}, \sigma^{ij}_{\text{what}} = e^{obj}_\phi(T(x, z^{ij}_{\text{where}})) \qquad z^{ij}_{\text{what}} \sim N(\mu^{ij}_{\text{what}}, \sigma^{ij}_{\text{what}})$$

Decoder Network. The decoder network is responsible for rendering the detected objects back into an image. First, an *object decoder network* d^{obj}_θ processes all z^{ij}_{what} to yield a reconstruction of the appearance of each object. Formally we have $o^{ij}, \alpha^{ij} = d^{obj}_\theta(z^{ij}_{\text{what}})$ where o^{ij} is an RGB volume with shape $(H_{obj}, W_{obj}, 3)$ (for fixed integers H_{obj}, W_{obj}), and α^{ij} is transparancy volume with shape $(H_{obj}, W_{obj}, 1)$.

Appearances of all objects are then stitched together to yield a final image, with z^{ij}_{where} used along with a spatial transformer to give the objects the correct size and location. α^{ij} is multiplied by z^{ij}_{pres} to ensure that objects with $z^{ij}_{\text{pres}} = 0$ are not drawn to the image. For objects that overlap spatially, z^{ij}_{depth} values are used to parameterize a convex combination between the objects, acting as a differentiable approximation of relative object depth. The output of rendering is an image x_{out}, which parameterizes $p_\theta(x|z)$ as a set of independent, pixel-wise Bernoulli variables.

Training. In the VAE framework the Evidence Lower Bound (ELBO) is given by:

$$\mathcal{L}(\theta, \phi) \coloneqq E_{z\sim q_\phi(z|x)}\left[\log p_\theta(x|z)\right] - D_{KL}(q_\phi(z|x) \parallel p(z)) \tag{1}$$

It can be shown that $\log p_\theta(x) \geq \mathcal{L}(\theta, \phi)$; thus, to train the network, a sample-based estimate of the ELBO is maximized with respect to ϕ and θ using gradient ascent (Kingma and Welling, 2013).

To backpropagate through the sampling process, we make use of the reparameterization trick (Kingma and Welling, 2013). For the Normally distributed random variables z^{ij}_{where}, z^{ij}_{what} and z^{ij}_{depth} this is straightforward. The discrete Bernoulli random variables z^{ij}_{pres} are replaced with Concrete variables, continuous relaxations of Bernoullis to which the reparameterization trick can be easily applied (Maddison et al., 2016). At validation and test time the Concretes are discretized via rounding.

Experiments

In this section we empirically demonstrate the advantages of SPAIR on a number of different tasks.

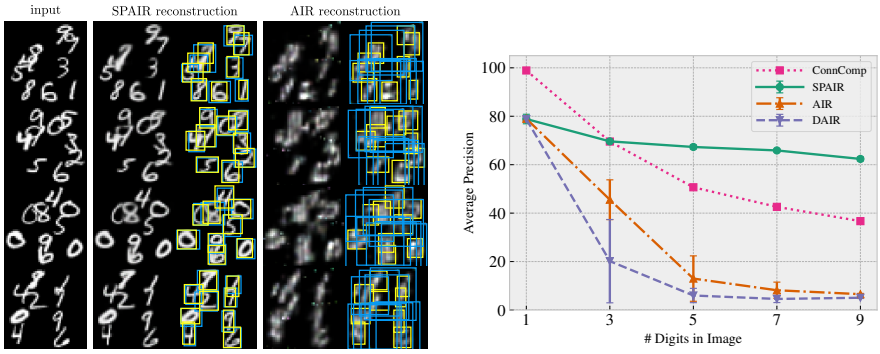

Figure 1: Left: Example input images and reconstructions by SPAIR and AIR. Predicted bounding boxes are shown in blue, ground-truth boxes in yellow. Right: Average Precision achieved by different algorithms on a scattered MNIST dataset as the number of digits per image varies.

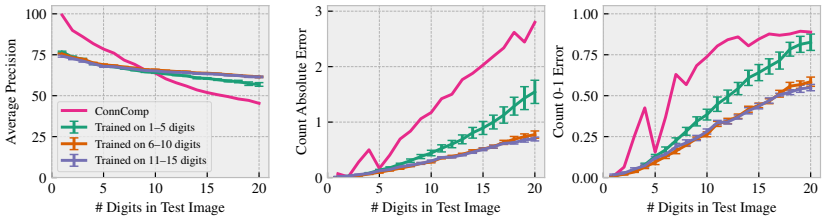

Figure 2: Assessing SPAIR's ability to generalize to images that are both larger and contain more objects than images seen during training. Left: Average Precision. The 6–10 and 11–15 lines overlap almost completely. Middle: Absolute difference between the number of objects predicted by the models and the true number of digits. Right: Mean 0-1 error between the number of objects predicted by models and the true number of digits.

Comparison with AIR. One of the main benefits that we expect to gain from SPAIR's spatial invariance is significantly improved ability to discover and detect objects in many-object scenes. To test this, we trained both AIR and SPAIR on $(48, 48)$ images each containing scattered MNIST digits of size $(14, 14)$. The goal is to have the models output accurate bounding boxes for the digits in each image, without ever having access to ground-truth bounding boxes. In order to probe the effect of the number of objects per image on model performance, we used 5 different training conditions; in each condition, the images contain a different number of digits, ranging from 1 to 9. As a performance measure we use an adapted version of the Average Precision between the bounding boxes predicted by the model and the ground-truth bounding boxes, commonly used in the supervised object detection literature (Everingham et al., 2010).

To simplify the comparison with AIR, we fixed the number of steps executed by AIR's recurrent network to the true number of objects in the image, effectively "telling" AIR how many objects are present. A variant of AIR called Difference Attend, Infer, Repeat (DAIR) (Eslami et al., 2016) was also tested and provided with this same information. We also constructed a simple baseline method which we call ConnComp which detects objects by finding connected components in an image. Success of this method can be used as a measure of the difficulty of the dataset: it will be successful to the degree that objects do not overlap and are easy to segment.

Results, shown in Figure 1, clearly demonstrate that on this task, SPAIR significantly outperforms all tested algorithms when the images contain many objects.

Generalization. Another hypothesized advantage of SPAIR's spatial invariance is a capacity for generalizing to images that are larger and/or contain different numbers of objects than images encountered during training. Here we test this hypothesis. We created three different training datasets, each consisting of small random crops (size $(48, 48)$) of larger images (size $(84, 84)$); the large images contain randomly scattered MNIST digits. In each training condition, the large pre-crop images contained different numbers of digits: either 1–5, 6–10 or 11–15 digits. To test generalization ability, the

models then were tested on large pre-crop images containing between 1 and 20 digits. In addition to AP, we also tracked how well the algorithms performed at guessing the number of objects in the scene. Results of this experiment, shown in Figure 2, demonstrate that SPAIR models have significant generalization ability. The performance of all SPAIR models degraded gracefully as the number of digits per test image increased, even well above the maximum number of digits seen during training. There is no significant difference between the performance of models trained on the 6–10 digit condition compared with models trained on the 11–15 digit condition. Models trained on the 1–5 digit condition exhibited lower performance when applied to images containing large numbers of digits, presumably because training experience did not equip them to deal with densely packed digits.

Space Invaders. To push the scaling capabilities of SPAIR further, we trained it on images from the Space Invaders Atari game using the Arcade Learning Environment (Bellemare et al., 2013), collected using a random policy. The network was trained on random crops of size $(48, 48)$, but at test time we had the network process full images, which have size $(210, 160)$. Qualitative results are given in Figure 3.

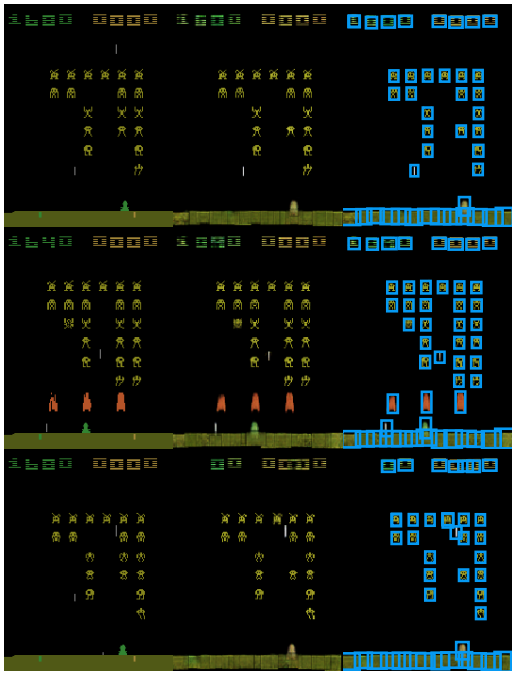

Figure 3: Example images from Space Invaders (left), and reconstructions (middle) and object bounding boxes (right) yielded by SPAIR.

Conclusion

In this paper, we introduced a novel architecture for unsupervised object detection which combines the unsupervised trainability of AIR with the spatial invariance properties of recent supervised object detection architectures such as YOLO. We showed empirically that of this spatial invariance allows for greatly improved scaling; in particular, we showed that our architecture outperforms competing approaches for images with many objects, and that it can generalize to images that are larger and more complex than images seen during training.

References

Bellemare, M. G., Naddaf, Y., Veness, J., and Bowling, M. (2013). The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279.

Chang, M. B., Ullman, T., Torralba, A., and Tenenbaum, J. B. (2016). A compositional object-based approach to learning physical dynamics. *arXiv preprint arXiv:1612.00341*.

Eslami, S. A., Heess, N., Weber, T., Tassa, Y., Szepesvari, D., Hinton, G. E., et al. (2016). Attend, infer, repeat: Fast scene understanding with generative models. In *Advances in Neural Information Processing Systems*, pages 3225–3233.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. (2010). The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338.

Jaderberg, M., Simonyan, K., Zisserman, A., et al. (2015). Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Maddison, C. J., Mnih, A., and Teh, Y. W. (2016). The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788.