Variational Saccading: Efficient Inference for Large Resolution Images

Jason Ramapuram[*,†]
Jason.Ramapuram@etu.unige.ch

Maurits Diephuis[*]
Maurits.Diephuis@unige.ch

Russ Webb[‡]
rwebb@apple.com

Alexandros Kalousis[*,†]
alexandros.kalousis@hesge.ch

Abstract

Image classification with deep neural networks is typically restricted to images of small dimensionality such as $\mathbb{R}^{224\times 244}$ in Resnet models [24]. This limitation excludes the $\mathbb{R}^{4000\times 3000}$ dimensional images that are taken by modern smartphone cameras and smart devices. In this work, we aim to mitigate the prohibitive inferential and memory costs of operating in such large dimensional spaces. To sample from the high-resolution original input distribution, we propose using a smaller proxy distribution to learn the co-ordinates that correspond to regions of interest in the high-dimensional space. We introduce a new principled variational lower bound that captures the relationship of the proxy distribution's posterior and the original image's co-ordinate space in a way that maximizes the conditional classification likelihood. We empirically demonstrate on one synthetic benchmark and one real world large resolution DSLR camera image dataset that our method produces comparable results with $\sim$**10x** faster inference and lower memory consumption than a model that utilizes the entire original input distribution.

1 Introduction

Direct inference over large input spaces allows models to leverage fine grained information that might not be present in their downsampled counterparts. We demonstrate a simple example of such a scenario in Figure 1, where the task is to identify speed limits. The downsampled image does not contain the required information to correctly solve the task; on the other hand direct inference over the original input space is memory and computationally intensive.

In order to work over such large dimensional input spaces, we take inspiration from the way the human visual cortex handles high dimensional input. Research in neuroscience [47, 23, 53] and attention for eye-gaze [54] have suggested that human beings enact rapid eye movements (or saccades [14]) to different locations within the scene to gather high resolution information from local patches. More recent research [26, 16] has shown that humans and macaque monkeys stochastically sample saccades from their environment and merge them into a continuous representation of perception. These saccades are also not necessarily only of the salient object(s) in the environment, but have a component of randomness attached to them. In this work we try to parallel this stochastic element through the use of a learnt sampling distribution, conditioned on auxiliary information provided via a proxy distribution. In this preliminary work we explore proxy distributions that are simply downsampled versions of the original large input distribution, as in Figure 1.

[*]University of Geneva
[†]Geneva School of Business Administration, HES-SO
[‡]Apple Inc

Third workshop on Bayesian Deep Learning (NeurIPS 2018), Montréal, Canada.

(a) **(b)**

Figure 1: **(a)** original image; **(b)** bilinearly downsampled image processable by a typical Resnet model.

2 Related Work

Saliency Methods: The analysis of salient (or interesting) regions in images has been studied extensively in computer vision [30, 29, 27, 21]. Important regions are quantified by simple low-level features such as intensity, color and orientation changes. These methods fail to generalize to complex scenes with non-linear relationships between textures and colors [7]. More recently, deep convolutional networks have been exploited to directly learn saliency at multiple feature levels (eg: [37, 8]) as well as to learn patch level statistics [56]. None of these methods directly learn "where" to look without information about the entire image.

CNN Approaches: Current state of the art CNN models on the other hand separate the *entire* image, into cropped regions [58], employ pyramid decompositions [38] over the *entire* image, or utilize large pooling [4] / striding operands. These methods are challenging because they are either lossy, resulting in poor classification accuracy, or they are too memory and computationally intensive (see Experiments Section 5) as they run convolutional filters over the entire image.

Region Proposal Methods: Another approach to CNN models are region proposal networks such as R-CNN [18], R-CNN++ [46] and YOLO [45] to name a few. The R-CNN methods generate a set of candidate extraction regions, either by extracting a fixed number of proposals as in the original work [18], or by utilizing a CNN over the *entire image* to directly predict the ROI [46]. They then proceed to enact a form of pooling over these regions, compute features, and project the features to the space of the classification likelihood. In contrast to R-CNN, our method uses an informatively learnt posterior to extract the exact number of required proposals, rather than the 2000 proposals as suggested in the original work. R-CNN++ on the other hand doesn't scale with ultra-large dimensional images as direct inference over these images scales with the dimensionality of the images. Furthermore, the memory usage of R-CNN++ increases with the dimensionality of the images whereas it does not for our proposed model.

YOLO on the other hand, resizes input images to $\boldsymbol{R}^{488\times488}$ and simultaneously predicts bounding boxes and their associated probabilities. While YOLO produces quick classification results, it trades off accuracy of fine-grained details. By resizing the original image, critical information can be lost (see Figure 1). Our proposed method on the other hand has no trouble with small details since it has the ability to directly control its foveation to sample the full resolution image.

Sequential Attention: Sequential attention models have been extensively explored through the literature, from utilizing Boltzman Machines [36, 13, 3], enacting step-by-step CNN learning rules [44], to learning scanning policies [1, 5] as well as leveraging regression based targets [25]. Our model takes inspiration from the recent Attend-Infer-Repeat (AIR) [15] and its extensions (SQAIR) [35], D.R.A.W [22], and Recurrent Attention Models (RAM) [40, 2]. While RAM based models allow for inference over large input images, they utilize a score function estimator [20] coupled with control variates [19]. Our algorithm on the other hand utilizes pathwise estimators [57, 34] which have been shown to have lower variance [52] in practice. In contrast to AIR and general attention based solutions, we do not use the entire image to build our attention map. In addition, as opposed to adding a classifier in an ad-hoc manner as in AIR and SQAIR, we derive a new principled lower bound on the conditional classification likelihood that allows us to relate the posterior of the proxy-distribution to the co-ordinate space of the original input. This direct use of supervised information in an end-to-end manner allows our model to converge very rapidly (100-300 epochs) vs AIR which takes 50,000-200,000 epochs [15] to successfully converge.

Interpretability: With the surge of deep learning, understanding the model decision making process has become more important. While prior work took a post-mortem approach on trained models by computing gradients of the conditional likelihood with respect to the input image [61, 39, 50, 41], recent work such as Capsule Networks [48], InfoGAN [11], and numerous others [48, 62, 59, 11] directly attempt to learn models that are interpretable[4]. Our model attempts to follow the latter of the two paradigms by extracting crops of regions that directly maximize the conditional classification likelihood. In contrast to the existing methods mentioned above we do not parse the entire input image to provide interpretability.

3 Variational Objective

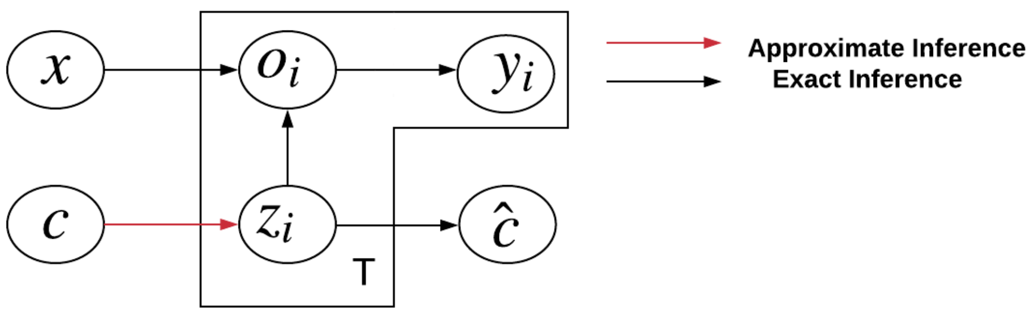


Figure 2: Graphical Model.

Given an image $\boldsymbol{x} \in \mathbb{R}^{K\times K}$, a corresponding proxy image $\boldsymbol{c} \in \mathbb{R}^{J\times J}$, $J \ll K$, and a corresponding class label $\boldsymbol{y} \in \mathbb{R}$, our objective is defined as maximizing $\log p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x})$ for $\boldsymbol{\theta}$. We are only interested in the case where $p(\boldsymbol{y}|\boldsymbol{c}) \neq p(\boldsymbol{y}|\boldsymbol{x})$, i.e. the proxy distribution is not able to solve the classification task of interest . Assuming that $\boldsymbol{c}$ provides no new information for the classification objective, $p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x}) = p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{c})$, and $p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}, \boldsymbol{x}) = p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})$, the conditional joint posterior $p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}, \boldsymbol{x})$ is only a function of the proxy distribution $\boldsymbol{c}$, we can reformulate our objective as:

$$\begin{aligned}\log p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x}) = \log p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{c}) &= \log \sum_{i=1}^{T} \int\int \Big(p_{\boldsymbol{\theta}_y}(\boldsymbol{y}_i|\boldsymbol{o}_i, \boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x}) \Big) d\boldsymbol{z}_i d\boldsymbol{o}_i \\ &= \log \sum_{i=1}^{T} \int\int \Big(p_{\boldsymbol{\theta}_y}(\boldsymbol{y}_i|\boldsymbol{o}_i, \boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x})\, p_{\boldsymbol{\theta}_o}(\boldsymbol{o}_i|\boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x})\, p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}) \Big) d\boldsymbol{z}_i d\boldsymbol{o}_i\end{aligned} \tag{1}$$

We have introduced (and marginalized out) two sets of latent variables: $\boldsymbol{z}_i \in \mathbb{R}^3$ and $\boldsymbol{o}_i \in \mathbb{R}^{L\times L}$, $L \ll J$. These correspond to the posteriors $\boldsymbol{z}_i \sim p(\boldsymbol{z}_i|\boldsymbol{c})$, induced by $\boldsymbol{c}$ and a set of dirac distributions, $\boldsymbol{o}_i \sim \delta[f(\boldsymbol{x}, \boldsymbol{z}_i)]$, centered at a differentiable function, f, implemented using Spatial Transformer networks [31]. This differentiable function produces crops, $\boldsymbol{o}_i$, of our large original input, $\boldsymbol{x}$, using a posterior sample from $p(\boldsymbol{z}_i|\boldsymbol{c})$. We utilize T such crops to approximate the discriminative regions of the image $\boldsymbol{x}$ that maximize the log-likelihood, $\log p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x})$.

To produce the crops $\boldsymbol{o}_i$, we utilize Spatial Transfomers (ST) [31]. STs transform the process of hard-attention based cropping (i.e. indexing into the image) with two differentiable operators: a learnt affine transformation of the *co-ordinate space* of the original image, $\begin{bmatrix} i^s & j^s \end{bmatrix}^T \mapsto \begin{bmatrix} i^t & j^t \end{bmatrix}^T$:

[4]See [63] for a more thorough treatment of interpretability.

$$\begin{bmatrix} i^t \\ j^t \end{bmatrix} = \begin{bmatrix} s & 0 & x \\ 0 & s & y \end{bmatrix} \begin{bmatrix} i^s \\ j^s \\ 1 \end{bmatrix} = \begin{bmatrix} z_0 & 0 & z_1 \\ 0 & z_0 & z_2 \end{bmatrix} \begin{bmatrix} i^s \\ j^s \\ 1 \end{bmatrix} \tag{2}$$

and a differentiable bilinear sampling operator that is independently applied on each channel c:

$$\sum_n^J \sum_m^J \left(x_{nm}^c \max(0, 1 - |i_{nm}^t - m|) \max(0, 1 - |j_{nm}^t - n|) \right)$$

In general the true posterior, $p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})$, is intractable or difficult to approximate [32]. To resolve this, we posit a variational approximation [55], $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}) \approx p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})$, and introduce it via a multiply-by-one constant:

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x}) = \log \sum_{i=1}^T \int \int \left(p_{\boldsymbol{\theta}_y}(\boldsymbol{y}_i|\boldsymbol{o}_i, \boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x})\, p_{\boldsymbol{\theta}_o}(\boldsymbol{o}_i|\boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x})\, p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}) \frac{q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})} \right) d\boldsymbol{z}_i d\boldsymbol{o}_i \tag{3}$$

Utilizing Jensen's inequality, we can reframe the marginalization operand as an expectation:

$$\begin{aligned} \log p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x}) &\geq \sum_{i=1}^T \int \int \left(q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}) \log \left[p_{\boldsymbol{\theta}_y}(\boldsymbol{y}_i|\boldsymbol{o}_i, \boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x})\, p_{\boldsymbol{\theta}_o}(\boldsymbol{o}_i|\boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x})\, \frac{p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})}{q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})} \right] \right) d\boldsymbol{z}_i d\boldsymbol{o}_i \\ &= \sum_{i=1}^T \int \Bigg[\mathbb{E}_{\boldsymbol{z}_i} \left(\log \left[p_{\boldsymbol{\theta}_y}(\boldsymbol{y}_i|\boldsymbol{o}_i, \boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x})\, p_{\boldsymbol{\theta}_o}(\boldsymbol{o}_i|\boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x}) \right] \right) \\ &\qquad - \mathrm{D}_{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}) || p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})] \Bigg] d\boldsymbol{o}_i \end{aligned} \tag{4}$$

We also observe that the KL divergence between the true posterior $p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})$ and the approximate posterior $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})$ can be re-written in terms of the Evidence Lower BOund (ELBO) [34] and the marginal data distribution $p(\boldsymbol{c})$:

$$-\mathrm{D}_{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}) || p_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})] = \mathbb{E}_{z_i}[\log p_{\boldsymbol{\theta}_c}(\hat{\boldsymbol{c}}|\boldsymbol{z}_i)] - \mathrm{D}_{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}) || p(\boldsymbol{z}_i)] - \log p(\boldsymbol{c}) \tag{5}$$

Given that $-\log p(\boldsymbol{c})$ is always a positive constant, we can update our reframed objective in Equation (4) by plugging in Equation (5):

$$\begin{aligned} \log p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x}) \geq \sum_{i=1}^T \sum_{\boldsymbol{z}_i} \sum_{\boldsymbol{o}_i} \Bigg(&\log p_{\boldsymbol{\theta}_y}(\boldsymbol{y}_i|\boldsymbol{o}_i, \boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x}) + \log p_{\boldsymbol{\theta}_o}(\boldsymbol{o}_i|\boldsymbol{z}_i, \boldsymbol{c}, \boldsymbol{x}) \\ &+ \log p_{\boldsymbol{\theta}_c}(\hat{\boldsymbol{c}}|\boldsymbol{z}_i) - \mathrm{D}_{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}) || p(\boldsymbol{z}_i)] \Bigg) \end{aligned} \tag{6}$$

This leads us to a Equation (6) which utilizes a empirical estimate of the expectation and marginalization operands to provide a novel lower bound on $\log p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{c})$. This lower bound allows us to classify a set of crops of the original image utilizing location information inferred by the posterior of the proxy distribution, $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})$. The equation presented in Equation 6 can also be extended to classification across a video sequence by replacing the variables $\{\boldsymbol{x}, \boldsymbol{c}\}$ with a set of variables $\{\boldsymbol{x}_i, \boldsymbol{c}_i\}_{i=1}^T$.

3.1 Interpretation

Current state of the art research in neuroscience for attention [26, 16] suggest that humans sample saccades approximately every 250ms and integrate them into a continuous representation of perception. We parallel this within our model by utilizing a discrete $\boldsymbol{o}_i \sim \delta[f(\boldsymbol{x}, \boldsymbol{z}_i)]$ for sampling saccades and continuous latent representations for the concept of perception. An additional requirement is the ability to transfer this continuous latent representation across glimpses. We overcome this barrier by utilizing a VRNN (Section 4).

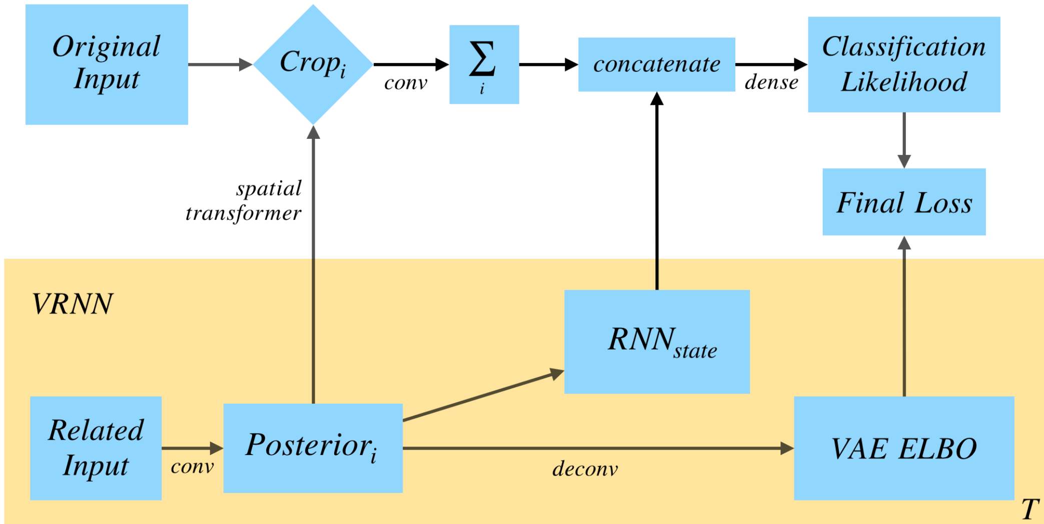


Figure 3: Implementation of our model.

In addition, [26, 16] show that attention does not always focus on the most salient object in an image, but at times randomly attends to other parts of the scene. This behavior can be interpreted as a form of exploration as done in reinforcement learning. Since our sampling distribution $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c})$ is stochastic, it provides a natural way to explore the space of the original input distribution $p(\boldsymbol{x})$, without the need for specific exploration methods such as ϵ-greedy [51] or weight noise [17].

4 Model

We use an isotropic gaussian VRNN [12] to model our newly derived lower bound (Equation 6). The VRNN makes two crucial modifications to the traditional ELBO: rather that assuming a non-informative prior, $p(\boldsymbol{z}_i)$ is learnt as a function of the previous RNN hidden state, $\boldsymbol{h}_{i-1}$, and the decoder, $p_{\boldsymbol{\theta}_c}(\hat{\boldsymbol{c}}|\boldsymbol{z}_i)$, and the encoder, $q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{z})$, are conditioned on the previous RNN hidden state :

$$\begin{aligned} p_{\boldsymbol{\theta}_p}(\boldsymbol{z}_i) &\sim \mathcal{N}(\boldsymbol{\mu}_i(\boldsymbol{h}_{i-1};\boldsymbol{\theta}_{\mu_p}),\ \boldsymbol{\sigma}_i^2(\boldsymbol{h}_{i-1};\boldsymbol{\theta}_{\sigma_p^2})) \\ p_{\boldsymbol{\theta}_c}(\hat{\boldsymbol{c}}|\boldsymbol{z}_i) &\sim \mathcal{N}(\boldsymbol{\mu}_i(g(\boldsymbol{z}_i),\boldsymbol{h}_{i-1};\boldsymbol{\theta}_{\mu_c}),\ \boldsymbol{\sigma}_i^2(f_p(\boldsymbol{z}_i),\boldsymbol{h}_{i-1};\boldsymbol{\theta}_{\sigma_c^2})) \\ q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{c}) &\sim \mathcal{N}(\boldsymbol{\mu}_i(f_q(\boldsymbol{z}_i),\boldsymbol{h}_{i-1};\phi_{\mu_q}),\ \boldsymbol{\sigma}_i^2(f_q(\boldsymbol{z}_i),\boldsymbol{h}_{i-1};\phi_{\sigma_q^2}))) \end{aligned}$$

This dependence on $\boldsymbol{h}_{i-1}$ allows the model to integrate and relay information about its previous saccade through to the next timestep. The full VRNN loss function is defined as:

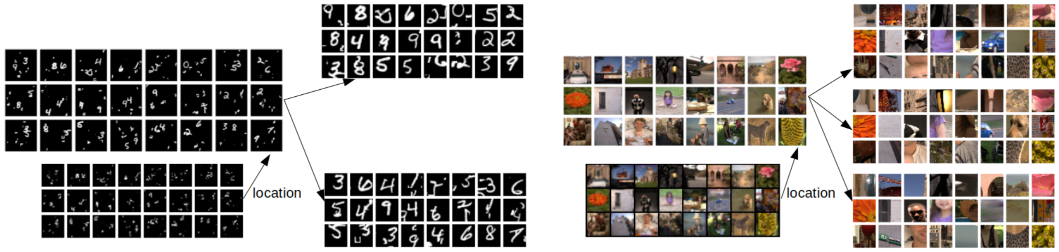
$$\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}_{\leq T}|\boldsymbol{c}_{\leq T})}\left(\sum_{i=1}^{T}\log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i|\boldsymbol{z}_{\leq i},\boldsymbol{x}_{<i}) - D_{KL}(q_{\boldsymbol{\phi}}(\boldsymbol{z}_i|\boldsymbol{x}_{\leq i},\boldsymbol{x}_{<i})||p(\boldsymbol{z}_i|\boldsymbol{x}_{<i},\boldsymbol{z}_{<i}))\right) \quad (7)$$

We implement the VRNN using a fully convolutional architecture where conv-transpose layers are used for upsampling from the vectorized latent space. The crop classifier is implemented by a standard fully-convolutional network, followed by a spatial pooling operation on the results of the convolution on the crops, $\boldsymbol{o}_i$. Adam [33] was used as an optimizer, combined with ReLU activations; batch-norm [28] was used for dense layers and group-norm [60] for convolutional layers. For more details about specific architectural choices see our code[5] .

5 Experiments

We evaluate our algorithm on two classification datasets where we analyze different induced behaviors of our model. We utilize Two-Digit MNIST for our first experiment in order to situate our model against baselines. We then proceed to learn a classification model for the large MIT-Adobe 5k dataset

[5]https://github.com/jramapuram/variational_saccading.git



(a) Top: original images; Bottom: proxy images
(b) Two test saccades of our model
(c) Top: original images; Bottom: proxy images
(d) Three test saccades of our model

Figure 4: *(a,b)*: Two-Digit-Identification ClutteredMNIST $\in \mathbb{R}^{2528\times2528}$; *(c,d)*: MIT-5k $\in \mathbb{R}^{2528\times2528}$.

Image Size: $\mathbb{R}^{2528\times2528}$	#params	gpu memory (160 batch-size)	time / epoch	accuracy MIT-Adobe-5k	accuracy Two Digit MNIST Identification
resnet18 - 144 crops	11.4M	79G (naive) 6.5G (checkpoint)[6]	1052.43s (naive) 1454.05s (checkpoint)	**63.6% +/- 0.03**	**97.3 +/- 0.006**
variational saccading	**7.4M**	**4.1G**	**120 s**	62.7% +/- 0.03	95.23 +/- 0.03

Image Size: $\mathbb{R}^{100\times100}$	#params	gpu memory (160 batch-size)	time / epoch	accuracy Two Digit MNIST Sum	accuracy Two Digit MNIST Identification
resnet18 - full image	11M	6.6G	59.27s	**99.86 +/- 0.01**	**97.4 +/- 0.003**
RAM [40]	-	-	-	91%	93%
DRAM [2]	-	-	-	97.5%	95%
variational saccading	**7.4M**	**2.8G**	**37s**	97.2 +/- 0.04	95.42 +/- 0.002

Table 1: Our model infers $\sim$ **9-10x** faster and utilizes less GPU memory than the baselines in high dimensions.

which involves complicated, dynamic, large resolution DSLR images. We utilize downsampled original images, $\boldsymbol{c} \in \mathbb{R}^{32\times32}$, as the proxy distribution for both experimentes. We demonstrate that our model has comparable accuracy to the best baseline models, but we infer $\sim$ **9-10x** faster and utilize far less GPU memory than a naive approach. We provide visualizations of the model's saccades; this aids in interpretting what region of the original input image aids the model in maximizing the desired classification likelihood. We utilize resnet18 as our naive baseline and did not observe any performance uplifts from using larger models for our experiments.

5.1 Two-Digit MNIST

Two-Digit-Cluttered MNIST is a benchmark dataset used in RAM [40], DRAM [2] and as a generative target in AIR [15] and SQAIR [35][7]. The objective with the initial set of experiments is to identity the digits present in the image (ignoring the distracting clutter), localize them, and predict a multi-class label using the localized targets. This form of learning, where localization information is not directly provided, is known as weakly supervised learning [9, 43, 42]. In the first setting we compare our model to RAM [40], DRAM [2] and a baseline resnet18 [24] model that operates over the entire image and directly provides classification outputs. As in RAM and DRAM, we also examine a case where the learning objective is to sum two digits placed in an image (without clutter). In order to provide a fair comparison we evaluate our model in the original dimension ($\mathbb{R}^{100\times100}$) suggested by the authors [40, 2]. We observe (Table 1 bottom) that our method is on par with RAM and DRAM and gets close to the baseline resnet18 results.

We extend the Two-Digit-Cluttered MNIST identification experiment from above to new experiment where we classify large dimensional images, $\boldsymbol{x} \sim \mathbb{R}^{2528\times2528}$. As in the previous experiment we evaluate our model against a baseline resnet18 model. Resnet models are tailored to operate over $\mathbb{R}^{224\times224}$ images; in order to use large images, we divide an original $\mathbb{R}^{2528\times2528}$ image into $\mathbb{R}^{144\times224\times224}$ individual crops and feed each crop into the model. We then sum the logit outputs of the model and run the pooled result through a dense layer. This allows the model to make a single classification decision for the entire image using all 144 crops:

[6]Checkpointing caches the forward pass operation as described in [10]. The naive approach parallelizes across 8 GPUs and splits each of the 144 crops across the GPUs.

[7]The authors do not use the cluttered version of the two-digit dataset for the AIR variants.

$$\boldsymbol{y} = f_{\boldsymbol{\theta}_d}(\sum_{i=1}^{144} g_{\boldsymbol{\theta}_c}(\boldsymbol{x}_i)) \ , \ \boldsymbol{x}_i \in \mathbb{R}^{224 \times 224} \tag{8}$$

In Equation 8, $f_{\boldsymbol{\theta}_d}$ is a multi-layer dense network and $g_{\boldsymbol{\theta}_c}$ is a multi-layer convolutional neural network that operates on individual crops $\boldsymbol{x}_i$. While it is also possible to also concatenate each logit vector $g_{\boldsymbol{\theta}_c}(\boldsymbol{x}) = [g_{\boldsymbol{\theta}_c}(\boldsymbol{x}_i), \ g_{\boldsymbol{\theta}_c}(\boldsymbol{x}_{i-1}), ..., \ g_{\boldsymbol{\theta}_c}(\boldsymbol{x}_0)]$, and project it through the dense network $f_{\boldsymbol{\theta}_d}(g_{\boldsymbol{\theta}_c}(\boldsymbol{x}))$, the tasks we operate over do not necessitate relational information [49] and pooled results directly aid the classification objective. We visualize saccades (Figure 4), the model accuracy, training-time per epoch and GPU memory (Table 1) and observe that our model performs similarly (in terms of accuracy) in higher dimensions, while inferring $\sim$**10x** faster and using **only 5%** of the total GPU memory in contrast to a traditional resnet18 model.

5.1.1 Robustness to Noisy Proxy Distribution

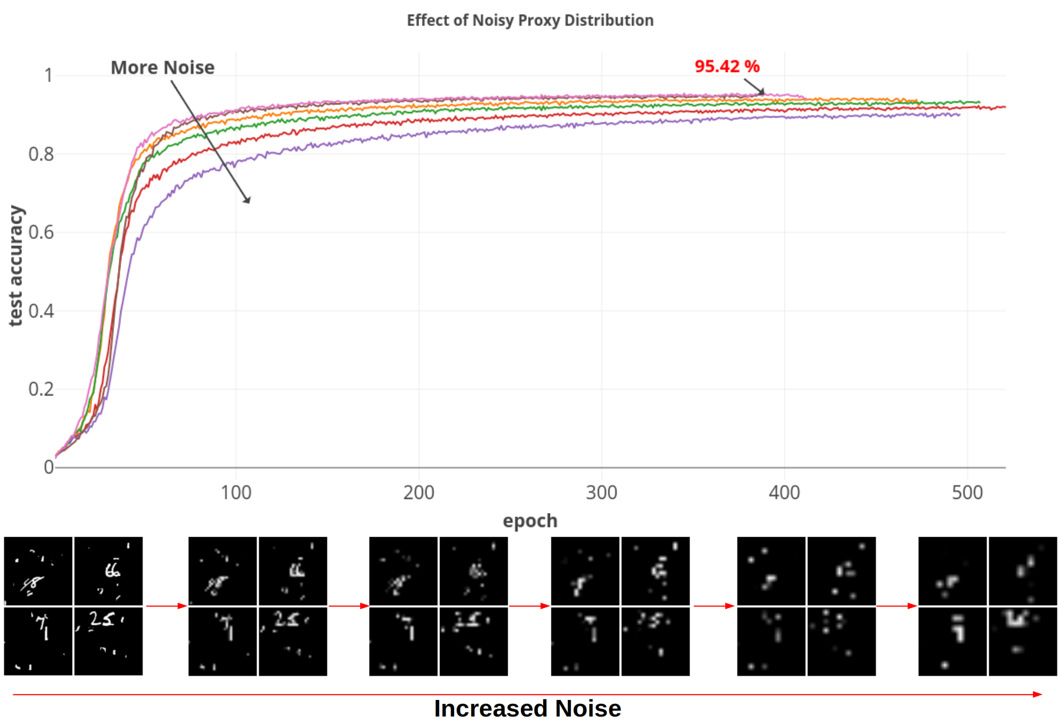


Figure 5: *Top*: Effect of noisy proxy distribution on test accuracy. *Bottom*: Left to right correspond to noisier versions of the same proxy distribution used in above graph.

Since the proxy distribution is critical to our formulation, we conduct an ablation study using the two-digit cluttered identification problem from experiment 5.1. We vary the amount of noise in the proxy distribution as shown in the *bottom* of Figure 5. The test curves shown on the *top* of the same figure demonstrates that our method is robust to noisy proxy distributions. In general, we found that our method worked even in situations where the proxy distribution only contained a few points, allowing us to infer positional information to index the original distribution $p(\boldsymbol{x})$.

5.2 MIT-Adobe 5k

The MIT-Adobe 5k [6] dataset is a large resolution DSLR camera dataset consisting of six classes: {abstract, animals, man-made, nature, None, people}. While the dimensionality of each image is large, the dataset has a total of 5000 total samples. This upper-bounds the performance of deep models with millions of parameters (without the use of pre-training / fine-tuning and other unsupervised techniques). We examine this scenario because it presents a common use case of learning in a low-sample regime.

We downsample the large original images to $\boldsymbol{x} \in \mathbb{R}^{3\times 2528\times 2528}$ to evaluate against a baseline resnet [24] model. The baseline model operates over 144 crops per image as in the previous experiment. Test saccades (non-cherry picked) of our model are visualized in Figure 4(c,d) ; the glimpses allow us to gain an introspective view into the model decision making process. Some of the interesting examples are that of the 'people' class: in the example with the child (third to the right in the bottom row of Figure 4), the model saccades to the adult as well as the child in the image. Other notable examples are leveraging the spotted texture of Cheetah fur and the snout of the dog. As observable from Table 1, our model has comparable accuracy to the baseline resnet model, but infers $\sim$ **9-10x** faster and utilize far less GPU memory than this naive approach.

6 Conclusion

We demonstrate a novel algorithm capable of working with ultra-large resolution images for classification and derive a new principled variational lower bound that captures the relationship of a proxy distribution's posterior and the original image's co-ordinate space . We empirically demonstrate that our model works with low memory and inference costs on ultra-large images using two datasets.

7 Acknowledgements

This research was supported by a H2020 FIRE+ initiative [8]. We also gratefully acknowledge the support of Amazon Inc via an AWS machine learning research grant that was used for this research.

References

[1] B. Alexe, N. Heess, Y. W. Teh, and V. Ferrari. Searching for objects driven by context. In *Advances in Neural Information Processing Systems*, pages 881–889, 2012.

[2] J. Ba, V. Mnih, and K. Kavukcuoglu. Multiple object recognition with visual attention. 2015.

[3] L. Bazzani, H. Larochelle, V. Murino, J.-a. Ting, and N. D. Freitas. Learning attentional policies for tracking and recognition in video with deep networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 937–944, 2011.

[4] Y.-L. Boureau, J. Ponce, and Y. LeCun. A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 111–118, 2010.

[5] N. J. Butko and J. R. Movellan. Optimal scanning for faster object detection. In *Computer vision and pattern recognition, 2009. cvpr 2009. ieee conference on*, pages 2751–2758. IEEE, 2009.

[6] V. Bychkovsky, S. Paris, E. Chan, and F. Durand. Learning photographic global tonal adjustment with a database of input / output image pairs. In *The Twenty-Fourth IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[7] Z. Bylinskii, A. Recasens, A. Borji, A. Oliva, A. Torralba, and F. Durand. Where should saliency models look next? In *European Conference on Computer Vision*, pages 809–824. Springer, 2016.

[8] F. Cao, Y. Liu, and D. Wang. Efficient saliency detection using convolutional neural networks with feature selection. *Information Sciences*, 456:34–49, 2018.

[9] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *IEEE transactions on pattern analysis and machine intelligence*, 29(3):394–410, 2007.

[10] T. Chen, B. Xu, C. Zhang, and C. Guestrin. Training deep nets with sublinear memory cost. *arXiv preprint arXiv:1604.06174*, 2016.

[11] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

[12] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio. A recurrent latent variable model for sequential data. In *Advances in neural information processing systems*, pages 2980–2988, 2015.

[13] M. Denil, L. Bazzani, H. Larochelle, and N. de Freitas. Learning where to attend with deep architectures for image tracking. *Neural computation*, 24(8):2151–2184, 2012.

[8] http://rawfie.eu

[14] H. Deubel and W. X. Schneider. Saccade target selection and object recognition: Evidence for a common attentional mechanism. *Vision research*, 36(12):1827–1837, 1996.

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

[16] I. C. Fiebelkorn, M. A. Pinsk, and S. Kastner. A dynamic interplay within the frontoparietal network underlies rhythmic spatial attention. *Neuron*, 99(4):842–853, 2018.

[17] M. Fortunato, M. G. Azar, B. Piot, J. Menick, M. Hessel, I. Osband, A. Graves, V. Mnih, R. Munos, D. Hassabis, et al. Noisy networks for exploration. *ICLR*, 2018.

[18] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[19] P. Glasserman. *Monte Carlo methods in financial engineering*, volume 53. Springer Science & Business Media, 2013.

[20] P. W. Glynn. Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM*, 33(10):75–84, 1990.

[21] S. Goferman, L. Zelnik-Manor, and A. Tal. Context-aware saliency detection. *IEEE transactions on pattern analysis and machine intelligence*, 34(10):1915–1926, 2012.

[22] K. Gregor, I. Danihelka, A. Graves, D. Rezende, and D. Wierstra. Draw: A recurrent neural network for image generation. In *International Conference on Machine Learning*, pages 1462–1471, 2015.

[23] M. Hayhoe and D. Ballard. Eye movements in natural behavior. *Trends in cognitive sciences*, 9(4):188–194, 2005.

[24] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[25] D. Held, S. Thrun, and S. Savarese. Learning to track at 100 fps with deep regression networks. In *European Conference on Computer Vision*, pages 749–765. Springer, 2016.

[26] R. F. Helfrich, I. C. Fiebelkorn, S. M. Szczepanski, J. J. Lin, J. Parvizi, R. T. Knight, and S. Kastner. Neural mechanisms of sustained attention are rhythmic. *Neuron*, 99(4):854–865, 2018.

[27] X. Hou and L. Zhang. Saliency detection: A spectral residual approach. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[28] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[29] L. Itti and C. Koch. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision research*, 40(10-12):1489–1506, 2000.

[30] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on pattern analysis and machine intelligence*, 20(11):1254–1259, 1998.

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

[32] D. P. Kingma. *"Variational Inference & Deep Learning: A New Synthesis"*. PhD thesis, 2017.

[33] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2014.

[34] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *ICLR*, 2014.

[35] A. Kosiorek, A. Bewley, and I. Posner. Hierarchical attentive recurrent tracking. In *Advances in Neural Information Processing Systems*, pages 3053–3061, 2017.

[36] H. Larochelle and G. E. Hinton. Learning to combine foveal glimpses with a third-order boltzmann machine. In *Advances in neural information processing systems*, pages 1243–1251, 2010.

[37] G. Li and Y. Yu. Visual saliency based on multiscale deep features. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5455–5463, 2015.

[38] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez. High-resolution image classification with convolutional networks. In *Geoscience and Remote Sensing Symposium (IGARSS), 2017 IEEE International*, pages 5157–5160. IEEE, 2017.

[39] A. Mahendran and A. Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5188–5196, 2015.

[40] V. Mnih, N. Heess, A. Graves, et al. Recurrent models of visual attention. In *Advances in neural information processing systems*, pages 2204–2212, 2014.

[41] C. Olah, A. Mordvintsev, and L. Schubert. Feature visualization. *Distill*, 2(11):e7, 2017.

[42] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 685–694, 2015.

[43] J. Ramapuram and R. Webb. A new benchmark and progress toward improved weakly supervised learning. *BMVC*, 2018.

[44] M. Ranzato. On learning where to look. *arXiv preprint arXiv:1405.5488*, 2014.

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

[46] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[47] R. A. Rensink. The dynamic representation of scenes. *Visual cognition*, 7(1-3):17–42, 2000.

[48] S. Sabour, N. Frosst, and G. E. Hinton. Dynamic routing between capsules. In *Advances in Neural Information Processing Systems*, pages 3856–3866, 2017.

[49] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. In *Advances in neural information processing systems*, pages 4967–4976, 2017.

[50] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[51] R. S. Sutton and A. G. Barto. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge, 1998.

[52] S. Tokui and I. Sato. Evaluating the variance of likelihood-ratio gradient estimators. In *International Conference on Machine Learning*, pages 3414–3423, 2017.

[53] A. M. Treisman and G. Gelade. A feature-integration theory of attention. *Cognitive psychology*, 12(1):97–136, 1980.

[54] S. K. Ungerleider and L. G. Mechanisms of visual attention in the human cortex. *Annual review of neuroscience*, 23(1):315–341, 2000.

[55] M. J. Wainwright, M. I. Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

[56] L. Wang, H. Lu, X. Ruan, and M.-H. Yang. Deep networks for saliency detection via local estimation and global search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3183–3192, 2015.

[57] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

[58] S. Wu, M. Zhang, G. Chen, and K. Chen. A new approach to compute cnns for extremely large images. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 39–48. ACM, 2017.

[59] T. Wu, X. Li, X. Song, W. Sun, L. Dong, and B. Li. Interpretable r-cnn. *arXiv preprint arXiv:1711.05226*, 2017.

[60] Y. Wu and K. He. Group normalization. *arXiv preprint arXiv:1803.08494*, 2018.

[61] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

[62] Q. Zhang, R. Cao, Y. N. Wu, and S.-C. Zhu. Growing interpretable part graphs on convnets via multi-shot learning. In *AAAI*, pages 2898–2906, 2017.

[63] Q.-s. Zhang and S.-C. Zhu. Visual interpretability for deep learning: a survey. *Frontiers of Information Technology & Electronic Engineering*, 19(1):27–39, 2018.