

Modeling Subitizing with Variational Autoencoders

*A bachelor thesis submitted to the University of Amsterdam,
presented for a bachelor's degree in Artificial Intelligence*

Rijnder Wever
10801944

Tom Runia (Supervisor)
PhD student in Computer Vision and Machine Learning
University of Amsterdam
Science Park 904, 1098 XH Amsterdam

18 EC Thesis



UNIVERSITY OF AMSTERDAM

July 6, 2018

Contents

1	Introduction	2
2	Related Work	4
3	Methods	5
3.1	Variational Autoencoder	5
3.1.1	Deep Feature Consistent Perceptual Loss	5
3.2	Hybrid Dataset	6
4	Experiments	7
4.1	Training the Variational Autoencoder	7
4.2	Hidden Representation Classifier	7
4.2.1	Classifier architecture	7
4.2.2	Class imbalance	8
5	Results & Discussion	8
5.1	Variational Autoencoder Performance	8
5.2	Subitizing Read-Out	9
5.3	Qualitative Analysis	10
6	Conclusion	11

Acknowledgements

Special thanks to Tom Runia, my bachelor thesis supervisor, for assisting me throughout this project and providing me with the necessary computing facilities. Furthermore, I want to thank the UvA RoboLab for granting me access to their prototyping machine. Finally, the Lisa Cluster provided me with all the computing power one could ever need, bringing peace of mind whenever I was worried I was using other's resources.

Modeling Subitizing with Variational Autoencoders

Rijnder Wever · Tom Runia

University of Amsterdam

July 6, 2018

Abstract Many animals develop the perceptual ability to *subitize*, allowing them to instantly and accurately assess the number of items in a small group. Such a visual number sense is shown to emerge in hierarchical generative models provided with spatial configurations of homogeneous rectangles in various sizes. A promising recent set of generative algorithms, *variational autoencoders* (VAEs), might allow training on more complex natural images, invariance to visual complexity being an essential aspect of any abstract sense of number. We show that our unsupervised VAE algorithm is able to spontaneously perform in subitizing judgments when provided with natural images from a dataset catered to subitizing. Although the VAE’s subitizing performance is behind some supervised algorithms, it encodes numerical information related to subitizing separately from other visual information, similar to the functioning of neuronal correlates of visual number sense. In particular, we show that the learned subitizing encoding is likely invariant to object area, comparable to how biological networks neglect area information when judging numerosity.

1 Introduction

Although various machine learning approaches dealing with the numerical determination of the amount of objects in images already exist, some research can be impartial to broader cognitive debate, resulting in models that are somewhat unhelpful for progressing the understanding of numerical cognition. Neurological findings about the general characteristics of cognitive processes performing in numerical tasks have successfully been applied to structuring and evaluating computational models of numerical cognition that aim to maintain biological plausibility (Stoianov and Zorzi, 2012; Zhang et al., 2016a). Essential to understanding the cognitive processes behind number sense is their perceptual origin, for so called *visual number sense* has been posed as the fundamental basis for developmentally later kinds of number sense, such as that required for arithmetical thinking and other more rigorous concepts of number found in mathematics (Lakoff and Núñez, 2000, chap. 2; Piazza and Izard, 2009). Visual number sense is the perceptual capability of many animals to immediately perceive a group of items as having either

a distinct or approximate cardinality, hereafter *numerosity* (Feigenson et al., 2004).

Numerosity-selective neurons have been shown to spontaneously emerge in dedicated neuronal networks, possibly as a result of numerosity being an integral part of the sensory world (Viswanathan and Nieder, 2013). As such, visual number sense can be said to emerge as a perceptual category in an interplay between numerosity-percepts and neural populations’ tuning to some preferred visual numerosity (Nieder, 2016). The experientially instantaneous capability to discriminate visual numerosity that many animals share (Nieder, 2016; Davis and Pérusse, 1988) can thus be explained by it primarily relying on a neuronal coding scheme, so avoiding a dependence on higher-cognitive processes such as conscious representations (Dehaene, 2011, p. 58 points to types of visual number sense being pre-attentive) or symbolic processing (visual numerosity percepts are understood non-verbally, Nieder, 2016). The response profile of neural populations detecting numerosity was found to be invariant to all visual features except quantity (Nieder, 2016; Harvey et al., 2013).

Similarly, computational models of numerical cognition provided only with sensory-like information show that an approximate visual sense of number can emerge as a property of hierarchically organized neural populations embedded in a generative learning model (Stoianov and Zorzi, 2012). An interesting biological motivation for such a model could be that biological numerosity detectors also exhibit hierarchical organization (Nieder, 2016; Viswanathan and Nieder, 2013), as well as aforementioned numerosity networks residing in the neo-cortex (Nieder, 2016), a brain area with an anatomical structure displaying likeness to hierarchical Bayesian generative models (Friston, 2003; although the brain in general is sometimes characterized as a probabilistic modeler of its sensory world, e.g. Knill and Pouget, 2004; Griffiths and Tenenbaum, 2006), resembling Stoianov and Zorzi (2012)’s model ability to distribute stochastic representations of sensory input across a multilayered, hierarchical network. Interestingly, the neurocognitively inspired model of Stoianov and Zorzi (2012) developed an encoding scheme that shared multiple similarities with the functional characteristics of biological neuronal populations performing in visual numerosity judgments, among

which an invariance to occupied visual area when judging numerosity.

Notwithstanding the success of previous biologically informed approaches to modeling numerical cognition with artificial neural networks, Stoianov and Zorzi (2012) note that more work is to be done in applying such models to natural images. The main reason behind pursuing natural images is improving the biological plausibility over previous approaches relying on binary images containing only simple geometric shapes (for examples, see Stoianov and Zorzi, 2012; Wu et al., 2018; Peterson and Simon, 2000), given that natural images are closer to everyday sensations than binary images. Furthermore, any dataset with visually uniform objects does not capture how visual number sense in animals is abstract in regard to the perceived objects (Nieder, 2016), implying that a model should be able to show that it performs equally well between objects of different visual complexities.

Unfortunately, no dataset fit for visual numerosity estimation tasks similar to Stoianov and Zorzi (2012) satisfied above requirements (sizable collections of natural images with large and varied, precisely labeled object groups are hard to construct), forcing present research towards *subitizing*, a type of visual number sense with a catered dataset readily available. Subitizing is the ability of many animals to immediately perceive the number of items in a group without resorting to e.g. enumeration or arithmetic, given that the number of items falls within the subitizing range of 1-4 (Kaufman et al., 1949; Davis and Pérusse, 1988). Aforementioned characteristics of visual sense of number hold equally well for more distinct types of numerosity perception such as subitizing. Similarly, subitizing is suggested to be a parallel pre-attentive process in the visual system (Dehaene, 2011, p. 57), the visual system likely relying on its ability to recognize holistic patterns for a final subitizing count (Jansen et al., 2014; Dehaene, 2011, p. 57; Piazza et al., 2002). This means that the “sudden” character of subitizing is caused by the visual system’s ability to process simple geometric configurations of objects in parallel, whereby increasing the size of a group behind the subitizing range deprives perceiving this group of its sudden and distinct numerical perceptual character for this would strain our parallelization capabilities too much. Groups exceeding the subitizing parallelization threshold force recourse to numerical judgment techniques with a less sudden perceptual character, such as counting by enumeration. Finally, above descriptions of subitizing have a supposed distinct neuronal correlate in the form of a capacity-limited multiple object individuation mechanism whose response profile obeys the perceptual subitizing range (Poncet et al., 2016), comparable to how the approximate numerosity system functions as an interplay between perception and coding properties of neurons.

Apart from the usage of natural images, present study will further improve biological plausibility by restricting the class of algorithm based on findings about visual sense of number. One such constraint is that an ability

to perceive numerosity is shown to emerge in hierarchically organized neural networks embedded in generative learning models, either artificial or biological (as discussed above). Furthermore, the fact that visual number sense exists across species (Davis and Pérusse, 1988), human newborns (Lakoff and Núñez, 2000, chap. 1) and innumerate cultures (Dehaene, 2011, p. 261; Franka et al., 2008) provides strong evidence for the claim that it is an implicitly developed skill at lower cognitive levels, given aforementioned groups lack of exposure to numerical training. Deemed as a general unrealistic trope of artificial learning (Dreyfus, 2007) and research into the human learning process (Zorzi et al., 2013a), modeling visual number sense therefore necessitates non-researcher depended features. This will restrict the choice of algorithm to so called *unsupervised* learning algorithms, as such an algorithm will learn its own particular representation of the data distribution. Given their ability to infer the underlying stochastic representation of the data in an unsupervised manner, i.e. perform in autonomous feature determination, *Variational Autoencoders* (VAEs) seem fit to tackle this problem, additional to VAEs being a generative algorithm (section 3.1 details their precise working). Moreover, VAEs are trained fully unsupervised similar to how, given appropriate circumstances, visual numerosity related abilities are implicitly learned skills that emerge without “labeled data”. Another interesting aspect of VAEs is their relatively interpretable and over-seeable learned feature space, which might tell something about how it deals with visual numerosity, and thus allows to evaluate the properties of the VAE’s encoding against biological data.

Present study therefore asks: how can neural networks provided with natural images be applied within biologically informed models to develop the emergent neuronal skill of subitizing in a manner comparable to their biological equivalents? To answer this, we start with discussing a dataset constructed for modeling subitizing in section 2, and then detail how we implemented our VAE algorithm to learn a representation of this dataset in 3.1. Slight problems during training and evaluation that arose from properties of our applied dataset and algorithm and appropriate solutions are examined in section 3.1.1, 3.2, and 4.2.2. Next, as the subitizing task is essentially an image classification task, a full methodology for evaluating the unsupervised VAE model’s performance on the subitizing classification task is described in section 4.2. Subitizing performance is reported in section 5.2. Finally, section 5.3 highlights measurements of the final models robustness to changes in visual features.

The contributions of our work are the following:

1. An unsupervised training setup in which a variational autoencoder algorithm spontaneously develops an ability to subitize is described. Our unsupervised approach seemed to require some specific countermeasures against problematic properties of the applied

subitizing dataset, namely class imbalance and data scarcity. Although their appliance proved only partly successful, further investigation could theoretically improve results.

2. We demonstrate that the performance of our unsupervised approach is comparable with supervised approaches using handcrafted features, although performance is still behind state of the art supervised machine learning approaches due to problems inherent to the particular VAE implementation.
3. Measuring the final models robustness to changes in visual features shows the emergence of a encoding property similar to biological neurons, that is to say, the VAE’s encoding scheme supports numerosity percepts invariant to visual features other than quantity.

2 Related Work

Visual Number Sense. Important characteristics of numerosity perception inspire algorithm architecture, evaluative measures and the training setup of present study. Firstly, visual sense of number is an *automatic* appreciation of the sensory world. It can be characterized as “sudden”, or as visible at a glance (Dehaene, 2011, p. 57; Zhang et al., 2016a). *Convolutional neural networks* (CNNs) not only showcase excellent performance in extracting visual features from complex natural images (Mnih et al., 2015; Krizhevsky et al., 2012; for visual number sense and CNNs see Zhang et al., 2016a), but are furthermore functionally inspired by the visual cortex of animals (specifically cats, see LeCun and Bengio, 1995). CNNs mimicking aspects of the animal visual cortex thus make them an appropriate candidate for modeling biologically plausible automatic neural coding of numerosity percepts. Secondly, the directness of visual numerosity entails that it does not require the interposition of external processes, or at least no other than lower-level sensory neural processes. For example, numerosity perception has been shown to function independently from mathematical knowledge of number, as numerosity detecting neuronal populations do not display their characteristic response profiles when confronted with Arabic numerals (Harvey et al., 2013; Poncet et al., 2016). This independence partly motivates the assessment of numerosity detectors solely encoding visual numerosity information. Visual number sense being an immediate and purely perceptual process implies that our model should not apply external computational techniques sometimes used in computer vision research on numerical determination task such as counting-by-detection (which requires both arithmetic and iterative attention to all group members, see Zhang et al., 2016a,b) or segmenting techniques (e.g. Chattopadhyay et al., 2016). Instead, we want the model to operate in an autonomous and purely sensory fashion.

Modeling approximate numerosity perception. Stoianov and Zorzi (2012) apply neural networks to learn-

ing visual numerosity estimation, although without using natural images. Stoianov and Zorzi (2012) discovered neural populations concerned with numerosity estimation that shared multiple properties with biological populations participating in similar tasks, most prominently an encoding scheme that was invariant to the cumulative surface area of the objects present in the provided images. Present study hopes to discover a similar kind of invariance to surface area. Likewise, we will employ the same surface area invariance test (section 5.3), although a successful application to natural images already implies a fairly abstract representation of number, as the objects therein are of varying visual complexity. Some simplicity of the dataset used by Stoianov and Zorzi (2012) is due their use of the relatively computationally expensive Restricted Boltzmann Machine (RBM) (except when exploiting prior knowledge of regularities in the probability distribution over the observed data, equation (20.6) from Goodfellow et al., 2016, shows that computational cost in RBMs grows as a multiple of the size of its hidden and observed units). Given developments in generative algorithms and the availability of more computational power, we will therefore opt for a different algorithmic approach (see section 3.1) that will hopefully scale better to natural images.

Subitizing Datasets. As seen in Fig. 1, the goal of the *Salient Object Subitizing* (SOS) dataset as defined by Zhang et al. (2016a) is to clearly show a number of salient objects that lies within the subitizing range. As other approaches often perform poor on images with complex backgrounds or with a large number of objects, Zhang et al. (2016a) also introduce images with no salient objects, as well as images where the number of salient objects lies outside of the subitizing range. The SOS dataset was constructed from an ensemble of other datasets to avoid potential dataset bias, and contains approximately 14K natural images. Zhang et al. (2016a) also report subitizing performance of multiple classification algorithms on images from the SOS dataset, although none are trained unsupervised.



Fig. 1: Example images from the SOS dataset. Each image shows a number of salient objects. The top row of images correspond to class labels 1-3, whereas the other two receive labels 0 and 4+.

3 Methods

3.1 Variational Autoencoder

VAEs (Kingma and Welling, 2013) are part of the family of autoencoder algorithms, owing this title to the majority of their structure consisting of an encoder and a decoder module (Doersch, 2016) (see Fig. 2 for the schematics of an autoencoder). In a regular autoencoder, the encoder module learns to map features from data samples $X \in \mathbb{R}^n$ into latent variables $z \in \mathbb{R}^m$ often so that $m \ll n$ and thus performs in dimensionality reduction, while the decoder function learns to reconstruct latent variables z into $X' \in \mathbb{R}^n$ such that X' matches X according to some predefined similarity measure (Liou et al., 2014). Reducing the input to be of much lower dimensionality forces the autoencoder to learn only the most emblematic regularities of the data, as these will minimize the reconstruction error. The latent space can thus be seen as an inferred hidden feature representation of the data.

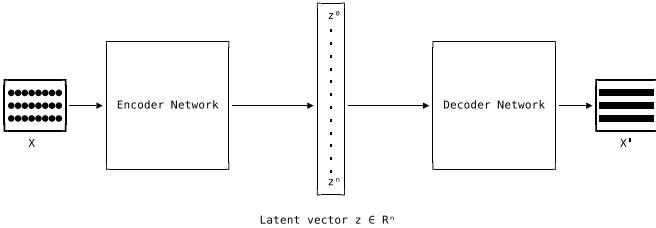


Fig. 2: Schematic architecture of an autoencoder. X indicates the input image, that is then encoded by the Encoder Network into a latent vector z . z is decoded back into a reconstruction X' of input X by the Decoder Network.

Where VAEs primarily differ from regular autoencoders is that rather than directly coding data samples into some feature space, they learn the parameters of a distribution that represents that feature space. Therefore, VAEs perform stochastic inference of the underlying distribution of the input data, instead of only creating some efficient mapping to a lower dimensionality that simultaneously facilitates accurate reconstruction. Now provided with statistical knowledge of the characteristics of the input, VAEs can not only perform reconstruction, but also generate novel examples that are similar to the input data based on the inferred statistics. The ability to generate novel examples makes VAEs a *generative* algorithm.

The task of the VAE's encoder network is to infer mean and variance parameters μ and Σ of a multivariate Gaussian distribution of the latent space, with μ and Σ arbitrary deterministic functions, such that latent vectors z drawn from this multivariate Gaussian $\mathcal{N}(\mu(X), \Sigma(X))$ facilitate reconstruction of data samples X (Doersch, 2016). Novel sampled z vectors can then be fed into the decoder network as usual. $\mu(X)$ and $\Sigma(X)$ are constrained to roughly follow a unit Gaussian by minimizing the Kullback-Leibler divergence (denoted as \mathcal{KL}) between

unit Gaussian $\mathcal{N}(0, I)$ and $\mathcal{N}(\mu(X), \Sigma(X))$, where \mathcal{KL} measures the distance between probability distributions. Normally distributed latent variables capture the intuition behind generative algorithms that they should support sampling latent variables that produce reconstructions that are merely *similar* to the input, and not necessarily accurate copies (Doersch, 2016). Furthermore, optimizing an arbitrary distribution would be intractable, thus VAEs need to rely on the fact that given a set of normally distributed variables $S = \{s_1, \dots, s_n\}$ with $S \in \mathbb{R}^n$ and any sufficiently complicated function $f(s_i)$ (such as a neural network), there exists a mapping $f : S \mapsto S'$ from which we can generate any arbitrary distribution $P(X) \in \mathbb{R}^n$ with $S' \sim P(X)$ (Doersch, 2016).

Therefore, the optimization objectives of a VAE become (Doersch, 2016):

1. \mathcal{KL} divergence loss \mathcal{L}_{kl} between the learned multivariate Gaussian and the unit Gaussian distributions, defined as:

$$\mathcal{L}_{kl} = \mathcal{KL}[\mathcal{N}(\mu(X), \Sigma(X)) || \mathcal{N}(0, I)]$$

2. Reconstruction loss \mathcal{L}_{rec} . Within visual problems, plain VAEs can for example minimize the binary cross entropy (BCE) between X and X' .

We define the total loss \mathcal{L}_{total} with weighting parameters α and β that weigh the relative important of each optimization objective:

$$\mathcal{L}_{total} = \alpha \mathcal{L}_{kl} + \beta \mathcal{L}_{rec} \quad (1)$$

Objective 1 grants VAEs the ability to generate new samples from the learned distribution, partly satisfying the constraint outlined in the introduction whereby visual numerosity related skills are shown to emerge in generative learning models. To fully satisfy this constraint, the final architecture uses deep convolutional networks for both the encoder and decoder module (see Fig. 4 for the VAE architecture), making the implementation a hierarchical model. The VAE is trained by reconstructing samples from the hybrid dataset (outlined in section 3.2) and SOS dataset until the loss converges. As a VAE's latent space encodes the most important features of the data, it is hoped the samples drawn from the encoder provide information regarding its subtitizing performance (see section 4.2).

3.1.1 Deep Feature Consistent Perceptual Loss

A BCE based reconstruction loss (see objective 2) yielded blurry reconstructions in our experiments, sometimes distorting numerical information by for example merging figure-ground organization. Because the frequently used pixel-by-pixel reconstruction loss measures in VAEs (such as a BCE based loss) do not necessarily comply with human perceptual similarity judgments, Hou et al. (2017) therefore propose optimizing the reconstructions (i.e. \mathcal{L}_{rec}) with help of the hidden layers of a pretrained

deep CNN, a particular advantage of these models being their ability to better capture spatial correlation relative to pixel-by-pixel measurements (Hou et al., 2017). For example, a pixel-by-pixel based measure yields a high difference between an image and its slightly shifted version as none of the pixels align anymore, while these two images will be deemed highly similar by humans. Additionally, CNNs have proven to model visual characteristics of images deemed important by humans, given their ability to for example perform complex image classification tasks (Krizhevsky et al., 2012), which should help our VAE in retaining the most important of information. The ability of the proposed *Feature Perceptual Loss* (FPL) to retain spatial correlation should reduce the noted blurriness (see for example Larsen et al., 2015) of the VAE’s reconstructions (Hou et al., 2017), which is especially problematic in subitizing tasks since blurring merges objects which in turn distorts subitizing labels. Hou et al. (2017) and present research employ VGG-19 (Simonyan and Zisserman, 2014) as the pretrained network Φ , trained on the large and varied ImageNet (Russakovsky et al., 2015) dataset. FPL requires predefining a set of layers $l_i \in L$ from pretrained network Φ , and works by minimizing the mean squared error (MSE) between the hidden representations of input x and VAE reconstruction \bar{x} at every layer l_i . Aside from the \mathcal{KL} divergence, the VAE’s reconstruction loss given L is now defined as:

$$\mathcal{L}_{rec}^L = \sum_{l \in L} \text{MSE}(\Phi(x)^l, \Phi(\bar{x})^l)$$

The intuition behind FPL is that whatever some hidden layer l_i of the VGG-19 network encodes should be retained in the reconstruction \bar{x} , as the VGG-19 has proven to model important visual characteristics of a large variety of image types. In Hou et al. (2017)’s and our experiments $L = \{\text{relu1_1}, \text{relu2_1}, \text{relu3_1}\}$ resulted in the best reconstructions.

3.2 Hybrid Dataset

We follow Zhang et al. (2016a) in pre-training our model with synthetic images and later fine-tuning on the SOS dataset. However, some small changes to their synthetic image pre-training setup are proposed. First, the synthetic dataset is extended with natural images from the SOS dataset such that the amount of instances per class is equal in every epoch (hopefully reducing problems encountered with class imbalance, see section 4.2.2). Another reason for constructing a hybrid dataset was the fact that the generation process of synthetic images was noted to 1. produce fairly unrealistic looking examples, and 2. be considerably less suitable than natural data for supporting subitizing performance (Zhang et al., 2016a). The second intuition behind this dataset is thus that the representation of the VAE must always be at least a little supportive of natural images, instead of settling on some optimum for synthetic images. The reason for including natural images in addition to synthetic data is that any tested

growth in dataset size during pre-training resulted into lower losses. The sampling ratio (detailed in section 4.1) of natural to synthetic images is increased at each epoch. We grow the original data size by roughly 8 times, pre-training with a total of 80000 hybrid samples per epoch. Testing many different parameters for the hybrid dataset was not given much priority as the total loss seemed to shrink with dataset expansion and training and testing a complete model was time expensive.

Synthetic images are generated by pasting cutout objects from THUS10000 (Cheng et al., 2015) onto the SUN background dataset (Xiao et al., 2010). The subitizing label is acquired by pasting an object N times, with $N \in [0, 4]$. For each paste, the object is transformed in equivalent manner to Zhang et al. (2016a). However, subitizing is noted to be more difficult when objects are superimposed, forcing recourse to external processes as counting by object enumeration (Dehaene, 2011, p. 57.), implying that significant paste object overlap should be avoided. Zhang et al. (2016a) avoid object overlap by defining a threshold $t \in [0, 1]$ whereby an object’s visible pixels $P_{visible}$ and total amount of pixels P_{total} should satisfy $P_{visible} > t * P_{total}$. For reasons given above, we define the object overlap threshold as $t = \sum_{n=0}^N 0.5 + n_i * 0.1$ with $N \in [0, 4]$ compared to Zhang et al. (2016a)’s static $t = 0.5$, as VAEs are especially prone to produce blurry reconstructions (Hou et al., 2017; Larsen et al., 2015), which requires extra care with overlapping objects as to not distort class labels. Refer to Fig. 3 for examples of generated synthetic images.



Fig. 3: Synthetic images generated for the pre-training stage. Images receive class labels 2, 3 and 4, from left to right. Although unrealistic looking, synthetic data generation counters class imbalance, displays a more extensive distribution of spatial configurations of objects, and familiarizes the model with a larger variety of object and background types. This is especially relevant for subitizing because it is abstract in regard to spatial information and the visual complexity of objects.

4 Experiments

4.1 Training the Variational Autoencoder

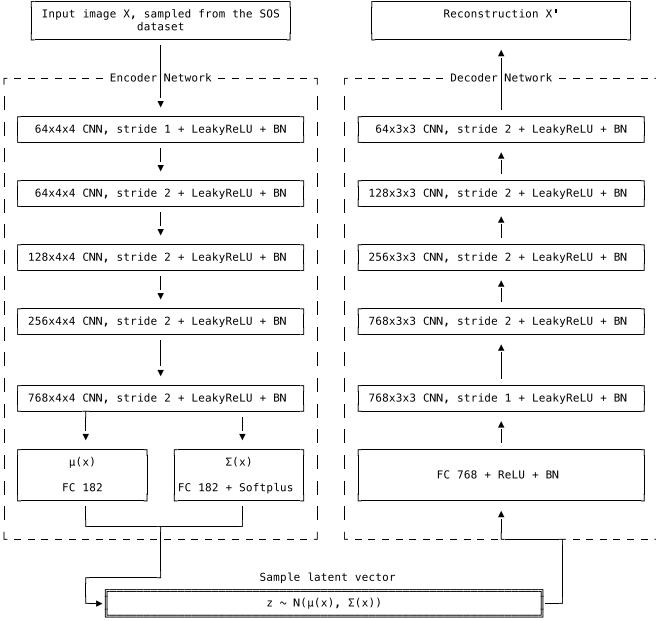


Fig. 4: Fully convolutional architecture of the encoder (left) and decoder (right) networks from the Variation Autoencoder. Note that this figure was simplified by omitting the reparameterization trick (Kingma and Welling, 2013), which enforces that latent samples are not drawn from $\mathcal{N}(\mu(x), \Sigma(x))$ directly. BN indicates a batch normalization layer (Ioffe and Szegedy, 2015) and FC indicates a fully connected layer. The dimensions before the CNN indicate filter and kernel size, respectively.

The fully convolutional (see section 2) architecture of our VAE is presented in Fig. 4. Subitizing results (Table 1) are reported on a second VAE with a slightly different architecture, for reasons outlined in section 5.2. The pre-training stage consist of feeding the VAE samples from the hybrid dataset (defined in section 3.2). The sampling ratio of natural to synthetic images within the hybrid dataset is increased at each epoch, and is defined by the Bézier curve shown in Fig. 5. After the loss converges on the hybrid dataset, the VAE is fine-tuned using only the SOS dataset by freezing the first three convolutional modules of the encoder network shown in Fig. 4. Data augmentation is performed on all sampled images, as the SOS dataset is relatively small in size (see section 4.2.2 and 5.1). Images are first resized to 183x183, after which a crop to 161x161 size is performed at a random location in the image. Additionally, each sample had 50% probability of being horizontally mirrored. Finally, the reconstructions of the VAE were observed to contain an abundance of primary color, so more complex color reconstruction was enforced by shifting each sample's R, G and B color channels (with values in range 0–255) by a small value drawn from a normal distribution with $\mu = 0$ and $\sigma = 10$ (the G channel uses $\sigma = 3.5$ because heavy green shifts produced unrealistic samples, possibly because green lighting contributes most to human color intensity perception Anderson et al., 1996). The loss weighting parameters from equation (1) are set as $\alpha = 1.0$ and $\beta = 0.03$. Although Hou et al. (2017) set β significantly closer to α , larger values of β were found to optimize FPL at the expense of the \mathcal{KL} divergence loss \mathcal{L}_{kl} under the SOS dataset. The size of the latent dimension is set to 182. The VAE was implemented¹ in PyTorch (Paszke et al., 2017).

255) by a small value drawn from a normal distribution with $\mu = 0$ and $\sigma = 10$ (the G channel uses $\sigma = 3.5$ because heavy green shifts produced unrealistic samples, possibly because green lighting contributes most to human color intensity perception Anderson et al., 1996). The loss weighting parameters from equation (1) are set as $\alpha = 1.0$ and $\beta = 0.03$. Although Hou et al. (2017) set β significantly closer to α , larger values of β were found to optimize FPL at the expense of the \mathcal{KL} divergence loss \mathcal{L}_{kl} under the SOS dataset. The size of the latent dimension is set to 182. The VAE was implemented¹ in PyTorch (Paszke et al., 2017).

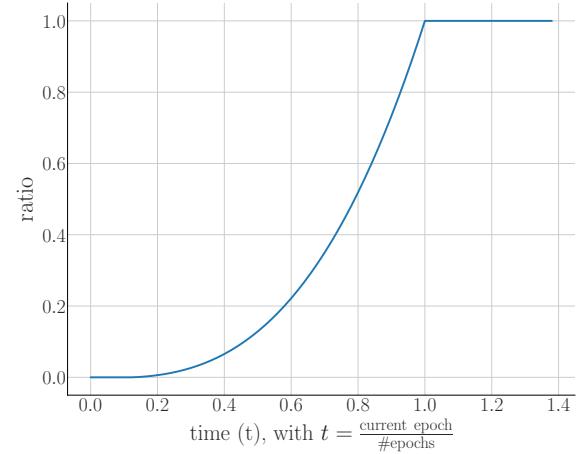


Fig. 5: Bézier curve with parameters $u_0 = 0, u_1 = -0.01, u_2 = 0.02, u_3 = 1.0$ defining the sampling ratio of natural images to synthetic images at time t .

4.2 Hidden Representation Classifier

4.2.1 Classifier architecture

To assess whether the learned latent space of the VAE showcases the emergent ability to perform in subitizing, a two layer fully connected network was fed with latent activation vectors z_{X_i} created by the encoder module of the VAE from an image X_i , and a corresponding subitizing class label Y_i , where X_i and Y_i are respectively an image and class label from the SOS training set. Both fully connected layers contain 160 neurons. Each of the linear layers is followed by a batch normalization layer (Ioffe and Szegedy, 2015), a ReLU activation function and a dropout layer (Srivastava et al., 2014). A fully connected network was chosen because using another connectionist module for read-outs of the hidden representation heightens the biological plausibility of the final approach (Zorzi et al., 2013b). Additionally, Zorzi et al. (2013b) note that the appended connectionist classifier module be conceived of as a cognitive response module supporting a particular behavioral task, although the main reason behind training

¹The code of the full implementation is hosted on [Github](#).

this classifier is to assess the VAE’s performance against other algorithmic data.

4.2.2 Class imbalance

Class imbalance is a phenomenon encountered in datasets whereby the number of instances belonging to one or more classes is significantly higher than the amount of instances belonging to any of the other classes. Although there is no consensus on an exact definition of what constitutes a dataset with class imbalance, we follow Fernández et al. (2013) in that given over-represented class c_m the number of instances N_{c_i} of one the classes c_i should satisfy $N_{c_i} < 0.4 * N_{c_m}$ for a dataset to be considered imbalanced. For the SOS dataset, $N_{c_0} = 2596$, $N_{c_1} = 4854$, $N_{c_2} = 1604$, $N_{c_3} = 1058$ and $N_{c_4} = 853$, which implies that c_0 and c_1 are *majority classes*, while the others should be considered *minority classes*. Most literature makes a distinction between three general algorithm-agnostic approaches that tackle class imbalance (for a discussion, see Fernández et al., 2013). The first two rebalance the class distribution by altering the amount of examples per class. However, class imbalance can not only be conceived of in terms of quantitative difference, but also as qualitative difference, whereby the relative importance of some class is weighted higher than others (e.g. in classification relating to malignant tumors, misclassifying malignant examples as nonmalignant could be weighted stronger than other misclassifications) Qualitative difference might be relevant to the SOS dataset, because examples with overlapping (i.e. multiple) objects make subitizing inherently more difficult (see section 3.2), and previous results on subitizing show that some classes are more difficult to classify than others (Zhang et al., 2016a).

1. *Oversampling techniques* Oversampling alters the class distribution by producing more examples of the minority class, for example generating synthetic data that resembles minority examples (e.g. He et al., 2008; Chawla et al., 2002), resulting in a more balanced class distribution.
2. *Undersampling techniques*. Undersampling balances the class distribution by discarding examples from the majority class. Elimination of majority class instances can for example ensue by removing highly similar instances. (e.g. Tomek, 1976)
3. *Cost sensitive techniques*. Cost sensitive learning does not alter the distribution of class instances, but penalizes misclassification of certain classes. Cost sensitive techniques are especially useful for dealing with minority classes that are inherently more difficult (or “costly”) to correctly classify, as optimization towards easier classes could minimize cost even in quantitatively balanced datasets if the easier classes for example require lesser representational resources of the learning model.

An ensemble of techniques was used to tackle the class imbalance in the SOS dataset. First, slight random undersampling with replacement of the two majority classes

(c_0 and c_1) is performed (see Lemaître et al., 2017), reducing their size by ~10%. Furthermore, as in practice many common sophisticated under- and oversampling techniques (e.g. data augmentation or outlier removal, for an overview see Fernández et al. (2013)) proved largely non-effective, a cost-sensitive class weighting was applied. The ineffectiveness of quantitative sampling techniques is likely to be caused by that in addition to the quantitative difference in class examples, there is also a slight difficulty factor whereby assessing the class of latent vector z is significantly more difficult if belongs to c_2 or c_3 versus any other classes, for these two classes require rather precise contours to discern individual objects, even more so with overlapping objects, while precise contours remain hard for VAEs given their tendency to produce blurred reconstructions (Larsen et al., 2015). The classifier network therefore seems inclined to put all of its representational power towards the easier classes, as this will result in a lower total cost, whereby this inclination will become even stronger as the quantitative class imbalance grows. The class weights for cost sensitive learning are set according to the quantitative class imbalance ratio (similar to section 3.2 in Fernández et al., 2013), but better accuracy was obtained by slightly altering the relative difference between the weights by raising them to some power n . In our experiments, $n = 3$ resulted in a balance between high per class accuracy scores and aforementioned scores roughly following the same shape as in other algorithms, which hopefully implies that the classifier is able to generalize in a manner comparable to previous approaches. For the SOS dataset with random majority class undersampling, if $n \gg 3$ the classifier accuracy for the majority classes shrinks towards chance, and, interestingly, accuracy for the minority classes becomes comparable to state of the art machine learning techniques.

5 Results & Discussion

5.1 Variational Autoencoder Performance

The VAE’s loss converged after 102 pre-training epochs, and once more after 39 epochs of fine-tuning solely on the SOS dataset. For the specific purpose of subitizing, we can see that using FPL loss is beneficial (indeed, when comparing FPL to BCE loss in the classification task described in section 4.2, the former performed better). For a comparison between FPL and BCE-based reconstruction measures on the SOS dataset, refer to Fig. 6. To get an idea of what sort of properties the latent space of the VAE encodes, see Fig. 7.

One notable shortcoming of FPL is that although the layers from VGG-19 represent important visual information, it is a known fact that the first few layers of deep CNNs only encode simple features such as edges and lines (i.e they support contours), which are only combined into more complex features deeper into the network (Liu et al.,



Fig. 6: Comparison between BCE and FPL reconstruction loss measures. The top row consist of original images from the SOS dataset, and the other two rows are reconstructions made by using FPL and BCE loss, respectively. The VAE using BCE shared the training setup of the VAE outlined in section 4.1, with the pre-training stage lasting 40 epochs, and the fine-tuning stage lasting 35 epochs. Some of these reconstructions support the claim that using FPL loss retains more numerical information.

2017; FPL’s authors Hou et al., 2017, note something similar). This means that the optimization objective is somewhat unambitious, in that it will never try to learn any other visual features (for examples, refer to Liu et al., 2017, Fig. 6.) aside from what the set of predefined layers L represents. Indeed, although contour reconstruction greatly improved with FPL, the reconstruction of detail such as facial features shows less improvement (Fig. 6). While Hou et al. (2017) show a successful application of FPL, they might have been unaware of this shortcoming due to using a more uniform dataset consisting only of centered faces.

Although the reconstructions show increased quality of contour reconstruction, there are a few reoccurring visual disparities between original and reconstruction. First of, novel patterns often emerge in the reconstructions, possibly caused by an implementational glitch, or a considerable difference in tested datasets (FPL is frequently paired with the CelebA Liu et al. (2015) dataset). Datasets other than the SOS dataset showed slightly better performance, indicating that the SOS dataset is either too small, too varied or requires non-standard tweaking for FPL to work in it’s current form. Most of the improvement in more uniform datasets came from the fact that the VAE learned to create local patterns to give the appearance of uniformly colored regions, but upon closer inspection placed colors in a grid such that they gave the appearance of just one color, similar to how for example LED screens function. Another reconstructional problem is that small regions such as details are sometimes left out, which could possibly distort class labels (for example, figure-ground contrast might decrease when detail is lost).

5.2 Subitizing Read-Out

Accuracy of the `zclassifier` (i.e. the classifier described in section 4.2.1 concerned with classification of latent activation patterns to subitizing labels) is reported in Table 1 over the withheld SOS test set. We report best performance using a slightly different VAE architecture than the one presented in Fig. 4 (which scored a mean accuracy of 40.4). The main difference between the VAE used in this experiment and the one that is used throughout the rest of this research is the placement of intermediate fully connected layers (with size 3096) between the latent representation and the convolutional stacks, as well as utilizing less convolutional filters. It is theorized that although the VAE with a fully convolutional architecture achieved a lower total loss and better reconstructional quality, a more complex model of the latent space does not necessarily facilitate an easier readout, and could possibly even degrade performance given the same classification algorithm (i.e. latent dimensions of the simpler VAE supporting in subitizing now also encode a variety of other information, as seen in Fig. 7, adding noise to the classification task).

Table 1: Average Precision (%) of various algorithms classifying subitizing labels to images from the SOS test set. Accuracy scores of algorithms other than our `zclassifier` were copied over from Zhang et al. (2016a). For their implementation, refer to Zhang et al. (2016a).

	0	1	2	3	4+	mean
Chance	27.5	46.5	18.6	11.7	9.7	22.8
SalPry	46.1	65.4	32.6	15.0	10.7	34.0
GIST	67.4	65.0	32.3	17.5	24.7	41.4
zclassifier	76.0	49.0	40.0	27.0	30.0	44.4
SIFT+IVF	83.0	68.1	35.1	26.6	38.1	50.1
CNN_FT	93.6	93.8	75.2	58.6	71.6	78.6

The subitizing performance of the VAE is comparable to the highest scoring non-machine learning algorithm, and performs worse overall than the CNNs trained by Zhang et al. (2016a). This can be explained by a number of factors. First of all, the `CNN_ft` algorithm used by Zhang et al. (2016a) has been pretrained on a large, well tested and more varied dataset, namely ImageNet (Russakovsky et al., 2015), which contains ≈ 1300 x more images than the SOS dataset. Additionally, their model is capable of more complex representations due to its depth and the amount of modules it contains (the applied model from Szegedy et al., 2015, uses 22 modules compared to the 12 in our approach). Moreover, all their algorithms are trained in a supervised manner, providing optimization algorithms such as stochastic gradient descent with a more directly guided optimization objective, an advantage over present research’s unsupervised training setup.



Fig. 7: Reconstructions of the image in the top-left, created by slightly increasing the response value of the VAE’s latent representation z at different individual dimensions z_i . Some dimensions give you a slight idea of what types of information they encode (e.g. a light source at a location)

5.3 Qualitative Analysis

Artificial and biological neural populations concerned with visual numerosity support quantitative judgments invariant to object size and, conversely, some populations detect object size without responding to quantity, indicating a separate encoding scheme for both properties (Stoianov and Zorzi, 2012; Harvey et al., 2013). Analogously, we tested whether our VAE’s latent representation contained dimensions z_i encoding either one of these properties. To test this, we first created a dataset with synthetic examples containing N objects ($N \in [0, 4]$, with N uniformly distributed over the dataset) and corresponding cumulative area values A that those N objects occupied (measured in pixels, with A normally distributed over the dataset²). The object overlap threshold was set as $t = 1$ for each example, to reduce noise induced by possible weak encoding capabilities and reasons outlined in section 3.2. As visualizations showed that each dimension z_i encodes more than one type of visual feature (see Fig. 7) special care was undertaken to reduce z_i ’s response variance by only generating data with 15 randomly sampled objects from the object cut-out set, and one random background class from the background dataset (performance is reported on the “sand deserts” class, which contains particularly visually uniform examples). A dimension z_i is said to be able to perform as either a numerical or area detector when regressing it’s response over the novel synthetic dataset ($n = 33280$) supports the following relationship between normalized variables A and N (Stoianov and Zorzi, 2012):

$$z_i = \beta_1 \log(N) + \beta_2 \log(A) + \varepsilon \quad (\text{with } N \in [0, 4]) \quad (2)$$

The regression was accomplished with linear regression algorithms taken from Newville et al. (2016) (Levenberg–Marquardt proved best). The criteria set by Stoianov and Zorzi (2012) for being a good fit of (2) are **1.** the regression explaining at least 10% of the variance ($R^2 \geq 0.1$) and **2.** an “ideal” detector of some property should have a low ($|\beta_i| < 0.1$) regression coefficient for the complementary property. We slightly altered criteria **1** to fit our training setup. The complexity of the SOS dataset in comparison to the binary images used by Stoianov and Zorzi (2012) requires our model to encode a higher variety of information, meaning that any fit is going to have more noise as no dimension z_i has just one role (see Fig. 7 for an overview). Moreover, the synthetic data we use for the regression includes more complex information than the dataset used by Stoianov and Zorzi (2012). Nevertheless, we still found a small number of reoccurring detectors of A and N , with $R > 0.065 \pm 0.020$ (all z_i with $R > 0.033$ resulted in an average $R = 0.07 \pm 0.015$). Due to randomization in the fitting process (synthetic examples are randomly generated at each run) the role distribution varied slightly with each property being encoded by about 1-2 dimensions, out of the total of 182 (anymore would indicate an unlikely redundancy, given that the small latent space should provide an efficient encoding scheme). Some latent dimensions that provide a better fit of (2) exist, but don’t satisfy criteria **2**. An interesting note is that whenever the regression showed multiple dimensions encoding area, they either exhibited positive or negative responses (i.e. positive or negative regression coefficients)

²An uniform distribution of cumulative area might have worked better, but required algorithmic changes to the synthetic data generation process that were inhibited by the available time.

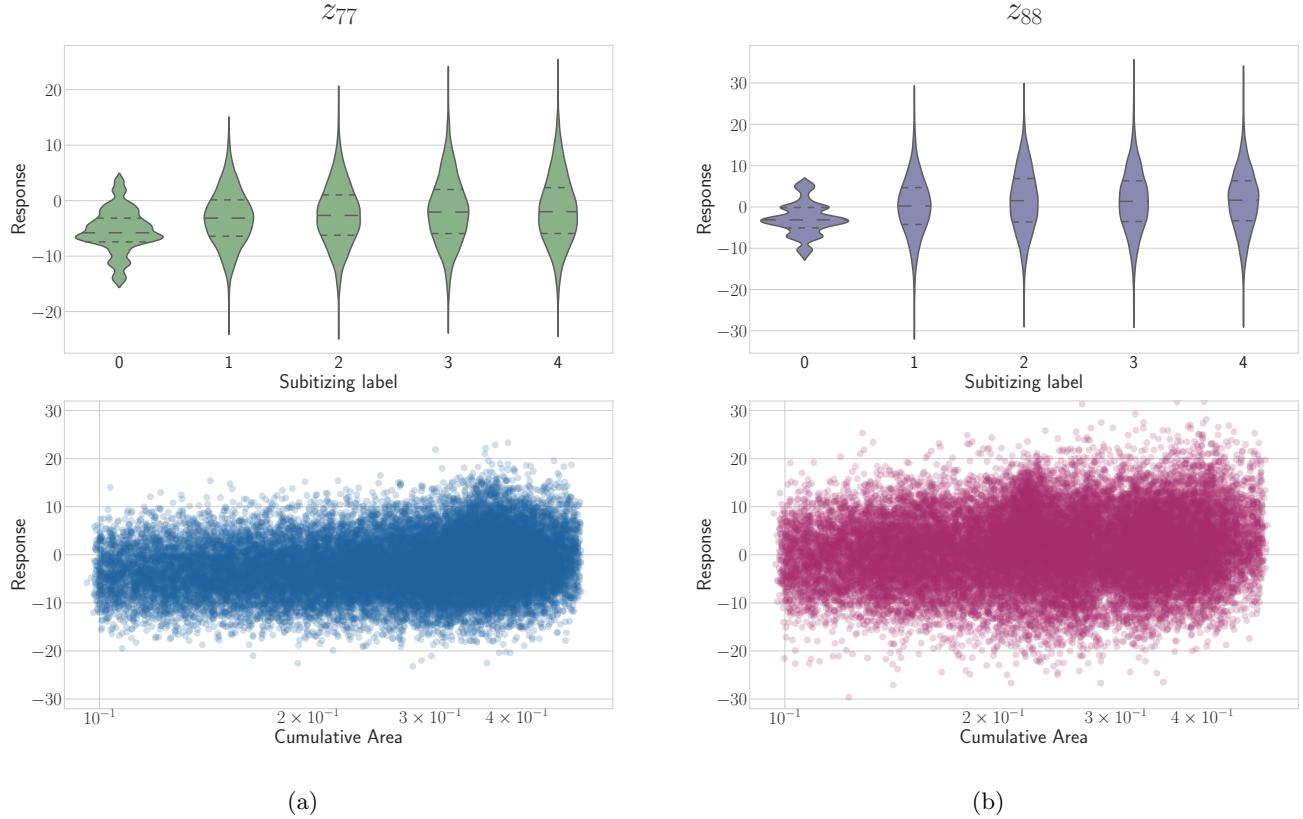


Fig. 8: (a) shows a typical response profile for a numerosity detector ($R = 0.055$). (b) shows a typical response profile of a dimension that encodes cumulative area while being invariant to numerosity information ($R = 0.056$). In the top violin plots, dotted lines indicate mean and variance of the response distribution, while thickness indicates frequency. Cumulative area (A) was normalized and is displayed across a logarithmic scale. For visual convenience, examples with $A = 0$ were shifted next to the lowest value of A in the dataset.

to area increase, in accordance with how visual numerosity might rely on a size normalization signal according to some theories on the neurocomputational basis for numerosity (for a discussion, see Stoianov and Zorzi, 2012). A large negative response (in contrast to a positive) to cumulative area might for example be combined with other responses in the VAE’s decoder network as an indicator or inhibitory signal that the area density does not come from just one object, but from multiple.

Fig. 8 provides characteristic response profiles for dimensions encoding either cumulative area or a subitizing count. For the area dimension (Fig. 8b), extreme cumulative area samples bend the mean distribution either upwards or downwards, while the response distribution to cumulative area for numerosity encoding dimensions stays relatively centered. The cumulative area detector z_{88} also shows an increasing response value relative to an increase in cumulative area, especially in comparison to z_{77} . For numerosity dimension z_{77} , Fig. 8a shows that both the total response and the center of the response distribution increased with numerosity. In contrast, the dimension that was sensitive to cumulative area shows a fairly static and discontinuous response to changes in subitizing count. With some extra time, the visual clarity

and overall performance of this qualitative analysis could probably be greatly improved, given that only a short focus on reducing response variance increased R by almost a factor of 10 in some cases.

6 Conclusion

We described a setup for training a VAE on a subitizing task, while satisfying some important biological constraints. A possible consequence thereof is that our final model showcases properties also found in biological neuronal networks (as well as other artificial algorithms). Firstly, an ability to subitize emerged as an implicitly learned skill. Second of all, the learned encoding scheme indicates support for encoding numerosities without resorting to counting schemes relying on cumulative object area, and conversely encodes cumulative area without using numerosity information, in accordance with findings in artificial models (Stoianov and Zorzi, 2012) and biological neuronal populations (Harvey et al., 2013; Nieder, 2016). However, more research is needed to assess the coding properties of area and numerosity detecting dimensions, given that some important properties of the

input to this task remain unexhausted, such as visual variation in synthetic images and the distribution of regression variables. There is also room for improvement in the VAE’s reconstruction abilities, i.e. efficiency of coding scheme, especially in regard to reconstruction of detail. A possible way to accomplish better coding of details is by experimenting with different optimization objectives (e.g. Dosovitskiy and Brox, 2016). An improved encoding might enhance performance in the subitizing classification task. Finally, other promising solutions to the class imbalance problem exist, such as dynamic cost-sensitive techniques, which were not implemented because the internal algorithmic changes they often require are time expensive. This class of algorithms looks promising because a static cost-sensitive approach yielded the best result out of all class imbalance countermeasures. Nevertheless, visual numerosity-like skills have emerged during the training of the VAE, showing the overall ability to perceive numerosity within the subitizing range without using information provided by visual features other than quantity. We can thus speak of a fairly abstract sense of number, as the qualitative analysis of the encoding yielded promising results over a large variation of images, whereby especially abstraction in regard to object area has been demonstrated.

References

- Ivilin Stoianov and Marco Zorzi. Emergence of a ‘visual number sense’ in hierarchical generative models. *Nature neuroscience*, 15(2):194, 2012.
- Jianming Zhang, Shuga Ma, Mehrnoosh Sameki, Stan Sclaroff, Margrit Betke, Zhe Lin, Xiaohui Shen, Brian Price, and Radomír Měch. Salient object subitizing. *arXiv preprint arXiv:1607.07525*, 2016a.
- George Lakoff and Rafael E Núñez. Where mathematics comes from: How the embodied mind brings mathematics into being. *AMC*, 10:12, 2000.
- Manuela Piazza and Véronique Izard. How humans count: numerosity and the parietal cortex. *The neuroscientist*, 15(3):261–273, 2009.
- Lisa Feigenson, Stanislas Dehaene, and Elizabeth Spelke. Core systems of number. *Trends in cognitive sciences*, 8(7):307–314, 2004.
- Pooja Viswanathan and Andreas Nieder. Neuronal correlates of a visual “sense of number” in primate parietal and prefrontal cortices. *Proceedings of the National Academy of Sciences*, 110(27):11187–11192, 2013.
- Andreas Nieder. The neuronal code for number. *Nature Reviews Neuroscience*, 17(6):366–382, may 2016. doi: 10.1038/nrn.2016.40. URL <https://doi.org/10.1038%2Fnrn.2016.40>.
- Hank Davis and Rachelle Pérusse. Numerical competence in animals: Definitional issues, current evidence, and a new research agenda. *Behavioral and Brain Sciences*, 11(4):561–579, 1988.
- Stanislas Dehaene. *The number sense: How the mind creates mathematics*. OUP USA, 2011.
- Ben M Harvey, Barrie P Klein, Natalia Petridou, and Serge O Dumoulin. Topographic representation of numerosity in the human parietal cortex. *Science*, 341(6150):1123–1126, 2013.
- Karl Friston. Learning and inference in the brain. *Neural Networks*, 16(9):1325–1352, 2003.
- David C Knill and Alexandre Pouget. The bayesian brain: the role of uncertainty in neural coding and computation. *TRENDS in Neurosciences*, 27(12):712–719, 2004.
- Thomas L Griffiths and Joshua B Tenenbaum. Optimal predictions in everyday cognition. *Psychological science*, 17(9):767–773, 2006.
- Xiaolin Wu, Xi Zhang, and Jun Du. Two is harder to recognize than tom: the challenge of visual numerosity for deep learning. *arXiv preprint arXiv:1802.05160*, 2018.
- Scott A Peterson and Tony J Simon. Computational evidence for the subitizing phenomenon as an emergent property of the human cognitive architecture. *Cognitive Science*, 24(1):93–122, 2000.
- E. L. Kaufman, M. W. Lord, T. W. Reese, and J. Volkmann. The discrimination of visual number. *The American Journal of Psychology*, 62(4):498, oct 1949. doi: 10.2307/1418556. URL <https://doi.org/10.2307%2F1418556>.
- Brenda RJ Jansen, Abe D Hofman, Marthe Straatemeier, Bianca MCW Bers, Maartje EJ Raijmakers, and Han LJ Maas. The role of pattern recognition in children’s exact enumeration of small numbers. *British Journal of Developmental Psychology*, 32(2):178–194, 2014.
- Manuela Piazza, Andrea Mechelli, Brian Butterworth, and Cathy J Price. Are subitizing and counting implemented as separate or functionally overlapping processes? *Neuroimage*, 15(2):435–446, 2002.
- Marlene Poncet, Alfonso Caramazza, and Veronica Mazza. Individuation of objects and object parts rely on the same neuronal mechanism. *Scientific reports*, 6:38434, 2016.
- Michael C Franka, Daniel L Everettb, Evelina Fedorenko, and Edward Gibson. Number as a cognitive technology: Evidence from pirahã language and cognitionq. *Cognition*, 108:819–824, 2008.

- Hubert L Dreyfus. Why heideggerian ai failed and how fixing it would require making it more heideggerian. *Philosophical psychology*, 20(2):247–268, 2007.
- Marco Zorzi, Alberto Testolin, and Ivilin P. Stoianov. Modeling language and cognition with deep unsupervised learning: a tutorial overview. *Frontiers in Psychology*, 4, 2013a. doi: 10.3389/fpsyg.2013.00515. URL <https://doi.org/10.3389%2Ffpsyg.2013.00515>.
- Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- Yann LeCun and et al. Bengio, Yoshua. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.
- Jianming Zhang, Stan Sclaroff, Zhe Lin, Xiaohui Shen, Brian Price, and Radomir Mech. Unconstrained salient object detection via proposal subset optimization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5733–5742, 2016b.
- Prithvijit Chattopadhyay, Ramakrishna Vedantam, Ramprasaath R Selvaraju, Dhruv Batra, and Devi Parikh. Counting everyday objects in everyday scenes. *arXiv preprint arXiv:1604.03505*, 2016.
- Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.
- Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- Carl Doersch. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*, 2016.
- Cheng-Yuan Liou, Wei-Chen Cheng, Jiun-Wei Liou, and Daw-Ran Liou. Autoencoder for words. *Neurocomputing*, 139:84–96, 2014.
- Xianxu Hou, Linlin Shen, Ke Sun, and Guoping Qiu. Deep feature consistent variational autoencoder. In *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*, pages 1133–1141. IEEE, 2017.
- Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. *arXiv preprint arXiv:1512.09300*, 2015.
- Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.
- Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):569–582, 2015.
- Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*, pages 3485–3492. IEEE, 2010.
- Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.
- Matthew Anderson, Ricardo Motta, Srinivasan Chandrasekar, and Michael Stokes. Proposal for a standard default color space for the internet—srgb. In *Color and imaging conference*, volume 1996, pages 238–245. Society for Imaging Science and Technology, 1996.
- Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.
- Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.
- Marco Zorzi, Alberto Testolin, and Ivilin Peev Stoianov. Modeling language and cognition with deep unsupervised learning: a tutorial overview. *Frontiers in psychology*, 4:515, 2013b.
- Alberto Fernández, Victoria López, Mikel Galar, María José del Jesus, and Francisco Herrera. Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches. *Knowledge-Based Systems*, 42:97–110, apr 2013. doi: 10.1016/j.knosys.2013.01.018. URL <https://doi.org/10.1016%2Fj.knosys.2013.01.018>.
- Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*, pages 1322–1328. IEEE, 2008.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

Ivan Tomek. Two modifications of cnn. *IEEE Trans. Systems, Man and Cybernetics*, 6:769–772, 1976.

Guillaume Lemaître, Fernando Nogueira, and Christos K. Aridas. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, 18(17):1–5, 2017. URL <http://jmlr.org/papers/v18/16-365>.

Mengchen Liu, Jiaxin Shi, Zhen Li, Chongxuan Li, Jun Zhu, and Shixia Liu. Towards better analysis of deep convolutional neural networks. *IEEE transactions on visualization and computer graphics*, 23(1):91–100, 2017.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3730–3738, 2015.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

Matthew Newville, Till Stensitzki, Daniel B Allen, Michal Rawlik, Antonino Ingargiola, and Andrew Nelson. Lmfit: non-linear least-square minimization and curve-fitting for python. *Astrophysics Source Code Library*, 2016.

Alexey Dosovitskiy and Thomas Brox. Generating images with perceptual similarity metrics based on deep networks. In *Advances in Neural Information Processing Systems*, pages 658–666, 2016.