A Joint Deep Neural Network and Evidence Accumulation Modeling approach to Human Decision-making with Naturalistic Images

William R. Holmes [1]
Department of Physics and Astronomy, Department of Mathematics, Quantitative Systems Biology Center, Vanderbilt University

Payton O'Daniels
Department of Computer Sciences, Vanderbilt University

Jennifer S. Trueblood
Department of Psychology, Vanderbilt University

Abstract

[1]E-mail: william.holmes@vanderbilt.edu

Evidence accumulation models (EAM) have proven to be an invaluable tool in probing the dynamical properties of decisions over recent decades. However, much of this literature has studied decisions utilizing simple stimuli where the experimenter has perfect knowledge and control over stimulus properties. Here we develop and test a new method for studying decisions involving naturalistic stimuli (medical images in this case) where the experimenter has neither perfect knowledge nor control of the stimuli properties. The central challenge in studying such decisions is to extract useful representations of images that can be associated with accumulation or drift rates in EAMs. Here we couple a deep convolutional neural network (CNN) with the diffusion decision model (DDM) to study how expert pathologists and novices make decisions involving the classification of digital images of blood cells as either normal (Non-Blast) or cancerous (Blast). In our approach, the CNN is the basis of a function that translates each image into a drift rate for use in the DDM. Results of fitting the joint CNN-DDM model to choice and response time data demonstrates that 1) both novices and experts demonstrated substantial speed accuracy tradeoffs, 2) both were susceptible to biases introduced by the presentation of pre-stimulus probabilistic cues, and 3) experts were more adept at extracting useful information from images than novices. These results demonstrate that this is a fruitful approach to studying decisions involving complex stimuli that will open new avenues for studying questions not possible with existing methods. Furthermore, this approach is technically feasible and has the potential to be translated into other domains of decision making research.

Keywords: diffusion decision model, convolutional neural network, medical image perception, Bayesian parameter estimation

Introduction

In many real life situations, individuals must make decisions based on complex visual information. These decisions range from deciding whether a weed growing in your garden is poisonous to a radiologist determining if a lung nodule is cancerous. In order to understand how these decisions are made and why errors sometimes occur, it is critical to understand how stimuli information influences the decision process.

Within the domain of simple perceptual decisions, several decades of research has shown that decisions are made through a process of evidence accumulation (Edwards, 1965; Ratcliff, 1978; Shadlen & Newsome, 1996; Gold & Shadlen, 2001; Smith & Ratcliff, 2004; Brown & Heathcote, 2008; Ratcliff & McKoon, 2008). That is, during the course of the deliberation, sensory evidence builds up for different responses. A choice is made once the accumulated evidence reaches an internally controlled threshold. While there are many

WRH and JST were supported by National Science Foundation Grant SES-1556325.

different computational instantiations of the evidence accumulation process, these models all assume that the rate at which evidence is accumulated is governed by the strength of stimulus information (modeled through a parameter called the drift rate). For example, consider the popular Random Dot Motion (RDM) discrimination task (Ball & Sekuler, 1982; Britten, Shadlen, Newsome, & Movshon, 1992, 1993) where participants view a cloud of dots, some of which move randomly and some of which move coherently, and are asked to choose the dominant direction of motion. In this paradigm, the drift rate is associated with the proportion of dots moving coherently (e.g., Hawkins, Forstmann, Wagenmakers, Ratcliff, & Brown, 2015). When the coherence level is low (e.g., only 10% of dots move in the same direction), the drift rate is small, reflecting the weak stimuli information. On the other hand, when the coherence level is high (e.g., 30% of dots move in the same direction), the drift rate is large, reflecting the strong stimuli information. More generally, in simple perceptual tasks, a different drift rate is typically associated with different stimuli (e.g., a RDM task with four coherence levels would have four drift rates). This approach allows modelers maximum flexibility to investigate the impact of stimuli strength on decision processes.

One of the current limitations in the application of evidence accumulation models to complex perceptual tasks (such as those involving naturalistic images) is linking stimulus information to the drift rate. In these cases, how does one translate each image into a numeric drift rate? In the present paper, we develop a framework for addressing this issue, and as an example, focus on the real world task of diagnosing cancer from images of white blood cells. In this task, the stimuli are a set of 300 unique images of white blood cells. Using the standard approach of allowing a separate drift rate for each stimulus would lead to a model with 300 drift rates, which is clearly intractable. An alternative approach is to break the images into classes and treat all images within each class as identical (as was done in Trueblood et al., 2018). In this prior study, the 300 images were grouped into four categories based on the image type (cancer or non-cancer) and difficulty (easy or hard) based on ratings from experts. Using these four categories, four drift rates were used in the modeling the data. Second, it requires extra information to determine the relevant classes; in this case, difficulty ratings from experts is required, which was time consuming to collect and might not be available for all tasks.

In the current paper, we propose an alternative way of addressing this issue for complex perceptual tasks involving naturalistic images. Specifically, we develop a deep convolutional neural network (CNN; eg., LeCun, Bengio, & Hinton, 2015) to provide a representation for each image. This representation is then used to translate each image into a drift rate where the numeric value of each drift rate is determined (using the CNN) by the characteristics of that image. This approach allows for maximum flexibility in incorporating stimuli information in an evidence accumulation model while retaining computational tractability.

This approach is also tractable and relatively simple to implement since it marries two well established quantitative analysis tools. First, pre-trained CNNs are lightly augmented and partially re-trained. The use of existing networks is of paramount importance since developing CNN's from scratch can be time consuming and require significant expertise. Performing only partial training of the CNN also speeds the process and reduces the need for vast numbers of training images, which is often not feasible. Second, the canonical

Diffusion Decision Model (DDM; Ratcliff, 1978) is used to model the decision making process (though any evidence accumulation model could in principle be used), with the CNN output serving as an input to the DDM. Numerous software implementations of the DDM have also been developed (e.g., Voss & Voss, 2007, 2008; Vandekerckhove & Tuerlinckx, 2007; Vandekerckhove, Tuerlinckx, & Lee, 2011). Thus existing models and software can be leveraged to implement this approach.

Models and Methods

In this paper, we develop and compare two models of medical image based decisions. One is a standard extension of the DDM that is included mainly for comparison purposes. The primary model being studied is a joint convolutional neural network (CNN) and DDM modeling platform to model decisions at the level of individual images. At a broad level, this model uses a partially custom CNN to assign a probability of being a cancer cell (termed a 'Blast' cell and denoted by P_{Blast}) to each image in the image bank (see Figure A1 for example images), which is then transformed into a drift rate to represent that image in the DDM. This joint CNN-DDM is then fit, at the level of individual trials, to choice and response time (RT) data using hierarchal Bayesian techniques. See Figure A1 for an overall schematic of this joint modeling approach. Below, we provide more details on this process.

The two models we discuss are both based on the standard DDM framework. The DDM predicts the distributions of choices and response times for a particular decision as a function of the strength of stimulus information (captured by the drift rate) and other parameters (mainly start point bias and threshold). The two models differ in how the drift rate is determined. In the primary model the drift rate associated with each image is determined by the output of the CNN. Thus in this model, each trial is independent, has its own associated drift rate, and no collapsing over trials is performed. From here on, we refer to this as the "CNN-DDM" model.

We compare this to a second, simpler model where images are categorized into classes as is commonly done. In addition to classifying each image in this data set as either Blast (cancer) or Non-Blast (non-cancer), a group of medical experts also provided an assessment of difficulty for each image. We used this second piece of information to classify each image as either "easy" or "hard". This yields four image classes based on the Blast / Non-Blast and Easy / Hard designations. In the second model, we collapsed over these four discrete classes and estimate a separate drift rate for each class. From here on, we will refer to this as the "Discrete Drift" model. Our purpose here is to compare the inferences one would make using this second, more standard modeling approach with the novel CNN based approach. Thus where relevant, we will directly compare parameter estimates (e.g., the diffusion model threshold) obtained with these two modeling approaches.

Diffusion decision model

Here we use a canonical diffusion decision model to investigate medical image based decisions. Specifically, we use a version that includes non-decision time (t_{ND}), start point bias (z), threshold (a), and a stimulus dependent drift rate (v). In the case of the Discrete Drift model, four separate drift rates will be estimated for the four image classes (e.g., hard Non-Blast). Importantly, in the CNN-DDM case, the drift rate (v_i) for image i is derived

from the characteristics of the medical image itself. That is to say, each individual image will have its own associated drift rate, which is determined by translating this image into a probability of being a Blast cell using the CNN approach described below. In this way, the characteristics of individual images, as determined by the CNN, determine the strength of evidence that forms the basis of the drift rate for that individual image. In the subsequent section, we discuss how this drift rate is determined and identify the model parameters associated with the drift rates that will be estimated. For simplicity and to make Bayesian hierarchal fitting of this data tractable, we do not include trial to trial noise variations in either start point or drift rate (typically referred to as sz and sv). In the CNN-DDM model, each image is treated as a single trial condition, and thus it is highly unlikely that these parameters would be estimable.

Translating images to probabilities using a deep CNN

To translate each medical image into a single numeric probability of being a Blast or non-Blast, we augmented a GoogLeNet deep CNN (Szegedy et al., 2015) that was pre-trained on the ImageNet database (we downloaded the fully pre-trained network). At a basic level this network (or any other CNN) consists of a sequence of layers that break images down into a feature vector (FV) followed by a sequence of layers that classify those images on the basis of those FVs. We took this pre-trained network and removed the classification layers, leaving only the layers that translate an image into a FV. We then added a single softmax classification layer that translates that FV into a probability of being a Blast cell (P_{Blast}). We chose a softmax layer specifically since its output can be interpreted as a probability of belonging to that class.

The following process was used to train this augmented network using "transfer learning". The pre-trained GoogLeNet was imported through Matlab's Deep Learning Toolbox using the Add-On Explorer. An image bank of 606 images (326 Blast and 280 NonBlast) were utilized. All images were pre-processed for use with this network by converting them to single precision arrays of size 224 pixels by 224 pixels. This image bank was broken into a training set (80%) and a validation set (20%). Matlab's Deep Learning Toolbox was used to train this network. In order to reduce the likelihood of overfitting, L2 regularization was used. The weights of the first 10 layers of the network were frozen (e.g. fixed at their pre-trained parameter values to reduce training time and prevent overfitting) and the newly added softmax layers were assigned a higher learning rate of 10 to facilitate their training. All intermediate layers were assigned the baseline learning rate of 1. Standard mini-batch stochastic gradient descent was used for training. After training of this network it had a classification accuracy of 94% on the validation set and 98% on the training set. This indicates the network exhibits adequate accuracy without overfitting, though the 4% gap indicates room for improvement, which is beyond the scope of this article. For further implementation details, see the Matlab Live Script (Matlab's equivalent of a Jupyter Notebook) that was used to train this network, which is available on the Open Science Framework at https://osf.io/j5hvp/.

At completion of training, the resulting network is used to extract classification probabilities for each image. This is then transformed into a log odds value via $LO = \log\left(P_{Blast}/P_{NonBlast}\right)$. For an individual image, the output log odds value is then used to construct a drift rate as described in the subsequent section.

Coupling the CNN with the diffusion model

The CNN described above takes each medical image and produces a measure of the stimulus information (i.e., the log odds) for that image. In particular, the LO provides both classification and strength information about each individual image. Negative (resp. positive) values correspond to Non-Blast (resp. Blast) images while values closer to 0 indicate less conclusive information while values further from 0 indicate more conclusive information. To connect the output of the CNN (LO) to the input of the DDM (drift), we utilize a linear function to map LO values into a drift rate, $v = v_{inter} + v_{slope} * LO$. Here, v_{inter} and v_{slope} are additional participant level model parameters that are estimated. v_{slope} in particular measures how adept participants are at translating visual information encoded in the images into evidence for either alternative. So for example we may expect this parameter to be smaller for novices than experts (see later results). Note that while we utilize a linear mapping from log-odds to drift rate in this application, in principle more complex mappings could potentially be included. For example, a saturating function that asymptotes at large log-odds values could be incorporated. There is no technical restriction on this functional from an implementation perspective. However it does lead to more parameters to estimate and a generally more complex model. Since the questions at hand do not warrant this added complexity, we have opted for simplicity in this application.

Hierarchal Bayesian parameter estimation

We fit two models, the CNN-DDM and discrete drift models, to choice-RT data in order to compare parameter estimates between the two and compare / contrast the inferences from them. As described in the Experimental Data section, the experiment consists of three conditions: Speed, Accuracy, and Bias. Furthermore, the images can be broken into four classes: easy Blast, easy Non-Blast, hard Blast, and hard Non-Blast.

Rather than fix certain parameters across conditions, we fit the speed, accuracy, and bias conditions completely separately for each model. The discrete drift model thus consists of seven parameters that are fit to each condition (eight in the case of the bias condition): threshold, start-point bias, non-decision time, and four drift rates corresponding to the four image classes. In the case of the bias condition, an additional start-point bias (one for cued conditions and one for non-cued conditions) is present. Thus the seven parameter model is fit separately to the speed and accuracy conditions and the slightly extended eight parameter model is fit to the bias condition.

The CNN-DDM is similarly fit to each instruction condition separately. It is a five (or six in the bias condition case) parameter model with the following parameters: threshold, start-point bias, non-decision time, and two parameters for the drift rate function (v_{inter} and v_{slope}). In the bias condition, there are once again two start-point bias parameters, one for the cue and another for the non-cued trial types. This five parameter model is fit to the speed and accuracy conditions separately and the slightly extended six parameter model is fit to the bias condition data.

In total, twelve model fits were performed. Specifically, the discrete drift model and CNN-DDM (two models) were fit to the separate conditions (three conditions) for both expert and novices (two experimental populations). In each case, we used Hierarchal Bayesian parameter estimation to estimate both group and individual level parameters

simultaneously. All figures depict posterior distributions for hyper mean parameters. The DEMCMC algorithm (Turner & Sederberg, 2014) was used. In order to efficiently calculate the DDM likelihood function, we used a variant of the algorithm in Navarro and Fuss (2009) where the short and long time series expansion times of the WFPT (Weiner First Passage Time) infinite series truncated at four terms (which yield errors $< 1e - 5$). The intra-trial variability parameter for the DDM was fixed at $s = 0.1$ as is common. See the Appendix for specification of the model priors.

Experimental Data

The data used for the modeling is from Trueblood et al. (2018), which examined the ability of novice undergraduate students and pathologists (residents and faculty) to distinguish between normal (standard white blood cells such as monocytes, lymphocytes, or neutrophils) and abnormal peripheral cells (blast cells, associated with acute leukemia) in clinical images.

Participants. 37 Vanderbilt University undergraduate students participated in the experiment for course credit. 19 pathologists from the Vanderbilt University Medical Center (VUMC) participated in exchange for a $15 gift card.

Materials. The stimuli were 300 digital images of Wright-stained white blood cells taken from anonymized patient peripheral blood smears at VUMC. The images were taken by an automated digital cell morphology instrument called the CellaVision DM96 (CellaVision AB, Lund, Sweden). Half of the images contained blast cells and the other half contained non-blast cells. In each category, half of the images were easy and half were hard. Thus, in total, there were 75 images in each of the four following categories: easy blast, hard blast, easy non-blast, and hard non-blast. These classifications were based on ratings from three hematopathology faculty from the Department of Pathology at VUMC. Details of the rating procedure and classification process can be found in Trueblood et al. (2018).

Procedure. All participants first completed training to familiarize themselves with blast cells. The main task consisted of six blocks with 100 trials in each block (25 images from each category). On each trial, participants were shown a single image and had to identify it as a blast or non-blast cell. There were three manipulations across blocks: accuracy, speed, and bias. In the accuracy blocks, participants were instructed to respond as accurately as possible and were given 5 seconds to respond. In the speed block, participants were instructed to respond quickly and were given 1 second to respond. In the bias blocks, participants were shown a probabilistic cue on half of the trials. The cue was a red dot that appeared before the image and identified the upcoming image as most likely being a blast cell. The cue was valid 65% of the time and participants were instructed about the validity. Full details of the procedures can be found in Trueblood et al. (2018).

Results

We first assess whether the CNN accurately classifies images in this data set. Subsequently, we fit the CNN-DDM and the discrete drift models to the choice-RT behavioral data and discuss results.

Validating the log odds output of the CNN as a measure of stimulus information

One of the primary goals of this approach is to use the CNN as a front end to the DDM to translate each image into a numeric value that represents the stimulus information in that image. Thus, first and foremost, the CNN must be able to accurately classify each image. We thus first determined the accuracy of classification. Figure A2a shows the classification accuracy for the hard and easy images respectively. Note that accuracy was measured on a set of test images that were not used in the training process, as is standard. Results show that indeed the CNN exhibits good accuracy. Furthermore, that accuracy is slightly lower on images determined by experts to be more difficult to classify.

In addition to measuring classification accuracy, we analyzed the distributions of output log odds values for the easy and difficult image classes (Figure A2b). Results show that, at a distributional level, easier images are assigned larger (in magnitude) LO values. In other words, easier images to classify are translated into a stronger strength of information while harder images are translated into a weaker strength of information. We caution that while this is true at the distributional level, it is not necessarily true for every image and some easy images will be assigned smaller LO values than other hard images as evidence by the overlap in LO distributions. It is possible that a fully purpose built CNN (e.g. a network that is designed and fully trained on the Blast cell classification task) may produce a stronger association between difficulty classification and LO assignment. This presents a number of technical challenges that are beyond the scope of this paper however.

In addition to producing a log-odds value to represent strength of information, each image can also be translated into a high dimensional (1024 dimensions in the case of GoogLeNet) feature vector. We next assessed whether Blast and Non-Blast images naturally cluster in this high dimensional feature vector space. To assess this, we utilized t-distributed stochastic neighbor embedding (tSNE, Maaten & Hinton, 2008) to visualize the cloud of feature vectors in this high dimensional space. Briefly, this is a non-linear dimension reduction technique that facilitates the visualization of high dimensional data. Results (Figure A2c) show that indeed the data set does cluster into two clusters of Blast and non-Blast images. Note that tSNE does not use the image labels in producing this embedding. The color coding is added after tSNE is applied. Thus the two clusters naturally appear in this data. This is expected since the CNN effectively classifies the images. None-the-less, it does raise the possibility of using these high-dimensional feature vector representations to calculate measures such as image similarity in future studies.

In conclusion, it is reasonable to use the LO output of this CNN as a measure of the stimulus type and strength for this task. At a binary level, it accurately assigns negative LO values to Non-Blast images and positive LO values to Blast images. Furthermore, at a distributional level it assigns larger LO values to easier images and smaller LO values to more difficult images. The CNN also naturally produces clusters representing Blast and Non-Blast image classes in feature vector space, which opens new possibilities for future studies use CNN based similarity measures in studies.

Model fitting and parameter estimation

We next fit the CNN-DDM and the discrete drift models to the behavioral data. Both models were fit to the two experimental populations (expert and novice) along with the three

experimental conditions separately, totaling six fits of each model. Figure A3 demonstrates the quality of fit for the CNN based model. Results show that model predictions of choice proportions are generally in good agreement with data as are mean response times for the speed an accuracy conditions. There is some discrepancy in the mean response times for the Bias condition. We however note that in these quantifications, we have calculated choice proportions and mean RTs for each of four categories: easy Blast, easy Non-Blast, hard Blast, and hard Non-Blast. This inherently assumes that there are distributions of choices and response times for each of these four conditions. While this collapsing over trials within a condition is common and reasonable in most scenarios (e.g those using standard DDM modeling where trials are grouped according to some commonality), trials were not grouped in any way when fitting the CNN-DDM. Rather, each trial was treated as an individual, distinct condition. Thus this collapsing is somewhat artificial and these comparisons should be interpreted with care. In the future, new methods that do not rely on these summary statistics need to be devised for this kind of modeling. We also note that the standard DDM was shown to fit this exact data well in Trueblood et al. (2018), and thus we do not duplicate those results here. In the figures that follow, violin plots depicting posterior distributions of the relevant hyper means are used to visualize model results.

Our intent here is not simply to provide a model based analysis of this medical decision task because this has been presented and analyzed previously in Trueblood et al. (2018). We are also not attempting to determine which model is "better" since they represent different approaches and require different data to be used (the discrete drift model requires difficulty data while the CNN-DDM does not). Rather, our purpose is to compare the conclusions one would draw from the two modeling approaches: the more standard discrete drift model and CNN-DDM model. We thus focus the discussion on comparison of parameter estimates between the models and conclusions that would be drawn from the different approaches.

Estimates for the threshold parameters (Figure A4) show two important results. First, applying time pressure in this medical context yields the standard effect of reducing the response threshold without altering other model parameters (see remaining figures). This is seen in both novices and experts using both the CNN-DDM and discrete drift models. More interestingly, the quantitative estimates of thresholds found with the CNN-DDM and discrete drift models are similar. Thus not only is the qualitative inference about the effect of time pressure the same for both models, the CNN-DDM yields similar parameter estimates for the threshold. Inspection of the bias parameter (Figure A5) and non-decision time parameter (Figure A6) show similar results; the two models yield similar parameter estimates.

While the drift parameters cannot be directly compared between the two models, analysis of those parameters yield similar conclusions as well. Comparison of drift rates between experts and novices in the discrete drift model (Figure A7) indicate that experts exhibit higher drift rates, likely due to the fact that they are more experienced and more adept at extracting useful information from these images. Inspection of the drift slope parameter from the CNN-DDM model (Figure A8) similarly shows that experts have a larger drift slope. Recall that drifts in the CNN-DDM are determined by $v = v_{inter} + v_{slope} * LO$. v_{slope} thus represents the extent to which stimulus information in the image is translated into evidence in the DDM process. A higher value of this parameter in experts indicates that they are better able to extract that stimulus information and utilize it in the decision

process.

Taken together these results suggest that this CNN-DDM approach yields the same qualitative inferences as the discrete drift model. Namely, that time pressure leads to a reduction in thresholds, the introduction of a cue leads to a slight start point bias, and that experts are more adept at translating the image characteristics into stimulus information. Furthermore, the quantitative estimates for threshold, bias, and non-decision time parameters, which the models share, are very similar.

General Discussion

Evidence accumulation models are one of the foundational tools on which our quantitative understanding of the temporal dynamics of decisions has been built. Much of the work in this domain however has centered on the study of decisions involving simple decision tasks such as the random dot motion task. In many real life situations however, decisions are based on complex visual information (medical images in the specific example presented here). Here we describe a new modeling approach that extends canonical EAMs to study decisions involving naturalistic stimuli.

One of the challenges in this domain is extracting a measure of strength of information from the stimuli, which is necessary to associate drift rates in EAMs with the stimuli. Unlike common, simple perceptual stimuli where the experimenter has precise knowledge (and control) of the stimuli characteristics, it is difficult to quantify the level of difficulty for naturalistic images (e.g., "how much" a given image looks like a blast or non-blast cell). In this new approach, we utilize a convolutional neural network to extract this information directly from the images themselves. In this way, the CNN serves as the basis of a function that translates images into drift rates.

In this study, we have shown that this approach is effective at yielding useful psychological inferences. First, these results demonstrate that this deep CNN can be effectively trained even on these complex medical image stimuli with as few as a few hundred images, which is a prerequisite for its further application. Second, with this CNN as the basis of a function that maps images into drift rates, the resulting joint CNN-DDM model can be fit to canonical choice and response time data using existing parameter estimation techniques (hierarchal Bayes in this case). Third, fitting this model to data to make parameter based inferences yields sensible conclusions. In particular, the addition of time pressure affects decision thresholds while the presentation of probabilistic cues yield alterations in the start-point bias parameters. This study thus validates this CNN-DDM approach and demonstrates that it yields reasonable psychological conclusions.

We do note that in this study we have made the rather strong assumption that the visual information extracted (e.g. feature vectors and log-odds values) from these images is identical for novices and expert pathologists. This, of course, is unlikely to be the case due to the extensive training experts receive. One avenue of future study would be to consider methods to augment the CNN element of this model to account for this. Two possibilities, for example, would be to assume that experts represent images with a higher dimensional feature vector than novices or to study the effects of adding noise to these feature vector to account for novices potentially having a noisier representation of image features.

This approach has a number of benefits over existing methods of modeling decisions with complex stimuli. First, it has the potential to simplify data collection and the design

of experiments. In our original study involving these images Trueblood et al. (2018), we had expert physicians rate images based on their difficulty. This is time consuming and requires experts, "difficulty" is an inherently subjective measure, and collection of such data may not always be possible. This approach thus frees one of the constraints of needing to collect this kind of preliminary data prior to performing a study. That said, some data curation is still required for this approach. Training a CNN requires labeled data and thus classification of a training data set (as Blast or non-Blast in this case) is required. However, with this CNN approach, it may not be necessary to provide expert classifications for the full data set (only enough for training is required) and difficulty data is no longer needed, both of which can simplify data collection.

It also opens up new possible avenues of study since decisions in this domain can now be studied at the level of individual choices for specific images. For example, does one image influence the perception of the next image, even though from a medical perspective they should be independent? Are errors confined to specific types of images and do these images share similar features? These and other questions are only possible with an approach such as the one described here.

Relation to prior work

Recent prior studies have also attempted to couple CNNs with cognitive models by using CNNs to produce representations of images that are then incorporated into the cognitive models. Annis and Palmeri (2018) coupled the output of a CNN to the Linear Ballistic Accumulator Model (Brown & Heathcote, 2008) in order to study object recognition. This study however utilized a network that received no training on the images of interest. While this may have been suitable for their study, we found that some training of the network in this medical task (as described in the methods) was absolutely required as the high dimensional feature vector representations of images that were generated by the naive CNN did not cluster at all (e.g. the gold and black points in Figure A2c completely overlapped). Additionally, our study went beyond asking whether the joint CNN - EAM could fit the data (which was the primary focus of Annis and Palmeri (2018)) and instead assessed the quality of the psychological conclusions obtained from the model. Sanders and Nosofsky (2018) alternatively coupled CNNs with the Generalized Context Model (Nosofsky, 1986) to study categorization. Our study differs from this in two primary ways. First, they performed substantially more manipulations of their CNN (they removed and added more CNN layers than we did). While this provides more flexibility, it also adds more complexity and we found it not to be necessary here. More significantly, they coupled the CNN with a different type of model (GCM versus EAMs) to study a different type of decision. While these studies in conjunction with that presented here investigate different types of decisions using different CNN implementations, they are similar in spirit and demonstrate that CNNs can provide adequate representations of image based stimuli and that there is a rich path forward in coupling CNNs with cognitive models to study decision making.

Feasibility, practicality, and extension to other decision tasks

This coupled CNN-EAM modeling approach can be readily applied in other domains involving image based decisions. This approach requires three elements to be carried out: 1) a decision task and image data set of interest, 2) a CNN to translate images into useful

numeric information, and 3) a EAM that describes the decision process being studied. With respect to the data, a labeled image set is required. That is, you must know what each image is for purposes of training the CNN. In our application, 800 images was sufficient to achieve $> 95\%$ accuracy. We tested the training of the CNN with $\sim$300 images as well and achieved accuracy of $\sim 87\%$. Thus large image sets are not required.

While CNNs can be challenging to design and computationally intensive to train, designing a novel network is not required for this application. The GoogLeNet used in this application was downloaded directly through Matlab (Python and other languages have similar implementations and capabilities) and all manipulation and training of the resulting augmented network were performed within Matlab. Furthermore, the training process itself took < 30 minutes on a standard desktop computer. We also note that GoogLeNet was chosen for convenience as it is relatively computationally efficient. We carried out the entire end to end modeling process described using a pre-trained ResNet50 (He, Zhang, Ren, & Sun, 2016) with identical results. Thus, at least in this case, the results appear to be independent of the CNN used. We have provided a Matlab script dedicated to the CNN training in order to serve as a starting point for anyone interested in utilizing this method (available on the Open Science Framework at https://osf.io/j5hvp/).

The ultimate purpose of this study is to use choice and RT data to probe the decision process. This required fitting an EAM to that data in order to extract model parameters from the data. While we utilized the diffusion decision model and custom software for this purpose, in principle any EAM model could be used (LBA for example Brown & Heathcote, 2008) and existing software packages could be utilized instead (such as the HDDM packageWiecki, Sofer, & Frank, 2013). In particular, recent development of the Probability Density Approximation (PDA) method for fitting complex models to data (Holmes, 2015; Turner & Sederberg, 2014; Holmes, Trueblood, & Heathcote, 2016; Holmes & Trueblood, 2018; Evans, Holmes, & Trueblood, 2019; Trueblood et al., 2018) opens up the possibility of investigating decisions using a range of different EAMs.

In conclusion, this approach is feasible and could be applied in a number of domains. Only relatively modest amounts of data are required. The design and training of CNNs is relatively simple and does not require significant expertise or computational capability. Finally, existing EAMs can be used, for which numerous efficient software packages exist.

Acknowledgements

We would like to thank Jeff Annis for helpful discussions regarding this research.

References

Annis, J., & Palmeri, T. (2018). Combining convolutional neural networks and cognitive models to predict novel object recognition in humans. *2018 Conference on Cognitive Computational Neuroscience*. doi: https://doi.org/10.32470/CCN.2018.1062-0

Ball, K., & Sekuler, R. (1982). A specific and enduring improvement in visual motion discrimination. *Science*, *218*(4573), 697–698.

Britten, K. H., Shadlen, M. N., Newsome, W. T., & Movshon, J. A. (1992). The analysis of visual motion: a comparison of neuronal and psychophysical performance. *The Journal of Neuroscience*, *12*(12), 4745–4765.

Britten, K. H., Shadlen, M. N., Newsome, W. T., & Movshon, J. A. (1993). Responses of neurons in macaque mt to stochastic motion signals. *Visual neuroscience*, *10*(06), 1157–1169.

Brown, S. D., & Heathcote, A. (2008). The simplest complete model of choice response time: Linear ballistic accumulation. *Cognitive psychology*, *57*(3), 153–178.

Edwards, W. (1965). Optimal strategies for seeking information: Models for statistics, choice reaction times, and human information processing. *Journal of Mathematical Psychology*, *2*, 312—329.

Evans, N. J., Holmes, W. R., & Trueblood, J. S. (2019, Feb 08). Response-time data provide critical constraints on dynamic models of multi-alternative, multi-attribute choice. *Psychonomic Bulletin & Review*. Retrieved from `https://doi.org/10.3758/s13423-018-1557-z` doi: 10.3758/s13423-018-1557-z

Gold, J. I., & Shadlen, M. N. (2001). Neural computations that underlie decisions about sensory stimuli. *Trends in Cognitive Sciences*, *5*(1), 10–16.

Hawkins, G. E., Forstmann, B. U., Wagenmakers, E.-J., Ratcliff, R., & Brown, S. D. (2015). Revisiting the evidence for collapsing boundaries and urgency signals in perceptual decision-making. *Journal of Neuroscience*, *35*(6), 2476–2484.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 770–778).

Holmes, W. R. (2015). A practical guide to the probability density approximation (pda) with improved implementation and error characterization. *Journal of Mathematical Psychology*, *68*, 13–24.

Holmes, W. R., & Trueblood, J. S. (2018, Apr 01). Bayesian analysis of the piecewise diffusion decision model. *Behavior Research Methods*, *50*(2), 730–743. Retrieved from `https://doi.org/10.3758/s13428-017-0901-y` doi: 10.3758/s13428-017-0901-y

Holmes, W. R., Trueblood, J. S., & Heathcote, A. (2016). A new framework for modeling decisions about changing information: The piecewise linear ballistic accumulator model. *Cognitive psychology*, *85*, 1–29.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436.

Maaten, L. v. d., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, *9*(Nov), 2579–2605.

Navarro, D. J., & Fuss, I. G. (2009). Fast and accurate calculations for first-passage times in wiener diffusion models. *Journal of Mathematical Psychology*, *53*(4), 222–230.

Nosofsky, R. M. (1986). Attention, similarity, and the identification–categorization relationship. *Journal of experimental psychology: General*, *115*(1), 39.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, *85*, 59-108.

Ratcliff, R., & McKoon, G. (2008). The diffusion decision model: Theory and data for two-choice decision tasks. *Neural Computation*, *20*(4), 873–922.

Sanders, C., & Nosofsky, R. (2018, July). Using deep-learning representations of complex natural stimuli as input to psychological models of classification. In J. Z. Chuck Kalish Martina Rau & T. Rogers (Eds.), *Cogsci 2018* (pp. 1025–1030).

Shadlen, M. N., & Newsome, W. T. (1996). Motion perception: Seeing and deciding. *Proceedings of the National Academy of Sciences*, *93*(2), 628–633.

Smith, P. L., & Ratcliff, R. (2004). Psychology and neurobiology of simple decisions. *Trends in neurosciences*, *27*(3), 161–168.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 1–9).

Trueblood, J. S., Holmes, W. R., Seegmiller, A. C., Douds, J., Compton, M., Szentirmai, E., ... Eichbaum, Q. (2018). The impact of speed and bias on the cognitive processes of experts and novices in medical image decision-making. *Cognitive Research: Principles and Implications*, *3*(1), 28.

Turner, B. M., & Sederberg, P. B. (2014). A generalized, likelihood-free method for posterior estimation. *Psychonomic bulletin & review*, *21*, 227-250.

Vandekerckhove, J., & Tuerlinckx, F. (2007). Fitting the ratcliff diffusion model to experimental data. *Psychonomic bulletin & review*, *14*(6), 1011–1026.

Vandekerckhove, J., Tuerlinckx, F., & Lee, M. D. (2011). Hierarchical diffusion models for two-choice response times. *Psychological methods*, *16*(1), 44-62.

Voss, A., & Voss, J. (2007). Fast-dm: A free program for efficient diffusion model analysis. *Behavior Research Methods*, *39*(4), 767–775.

Voss, A., & Voss, J. (2008). A fast numerical algorithm for the estimation of diffusion model parameters. *Journal of Mathematical Psychology*, *52*(1), 1–9.

Wiecki, T. V., Sofer, I., & Frank, M. J. (2013). Hddm: hierarchical bayesian estimation of the drift-diffusion model in python. *Frontiers in neuroinformatics*, *7*, 14.

Appendix
Model Priors

Two models were fit (to multiple data sets) in this article, the discrete drift DDM and the CNN-DDM. Here we describe the priors used for hierarchal Bayesian estimation of both models. Note that the DDM for this application specifies the thresholds to be $[0, a]$ rather than $[-a, a]$ and that the intra-trial variability parameter is set to $s = 0.1$. Both models contain four common parameters: threshold (a), bias (z), and non-decision time (t_{ND}). The same priors are used for these parameters in both models. The individual level priors were

$$a, z \sim TN(\mu_{a,z}, \sigma_{a,z}, 0, 2), \qquad , t_{ND} \sim TN(\mu_{ND}, \sigma_{ND}, 0.05, 1), \tag{1}$$

where $TN(\mu, \sigma, L, U)$ indicates the truncated normal with mean (μ), standard deviation (σ), lower bound (L), and upper bound (U). For these parameters, the hyper level parameters are distributed according to

$$\mu_{a,z} \sim TN(1, 0.5, 0, 10), \qquad \sigma_{a,z} = E(1), \tag{2}$$

$$\mu_{ND} \sim TN(0.15, 0.5, 0.05, 1), \qquad \sigma_{ND} = E(0.5), \tag{3}$$

where $E(\lambda)$ is the exponential distribution with decay parameter λ.

The models differ in their drift rate specifications. In the discrete drift rate model there are four drift parameters. The individual level priors are similar for each with

$$v \sim TN(\mu_v, \sigma_v, -2, 2), \tag{4}$$

where

$$\mu_v \sim TN(\pm 0.5, 0.5, -2, 2), \qquad \sigma_v = E(1), \tag{5}$$

and the $\pm$ indicates a plus sign for Blast stimuli and the negative sign for Non-Blast stimuli.

For the CNN-DDM, there are two drift parameters corresponding to the slope and intercept of the drift function. These are distributed according to

$$v_{slope} \sim TN(\mu_{slope}, \sigma_{slope}, 0, 10), \qquad v_{inter} = TN(\mu_{inter}, \sigma_{inter}, -10, 10). \tag{6}$$

The hyper parameters are then distributed according to

$$\mu_{slope} \sim TN(0.25, 1, 0, 10), \qquad \sigma_{slope} = E(1), \tag{7}$$

$$\mu_{inter} \sim TN(0, 1, -10, 10), \qquad \sigma_{inter} = E(1). \tag{8}$$

Priors are the same for all fits of the speed, accuracy, and bias conditions.

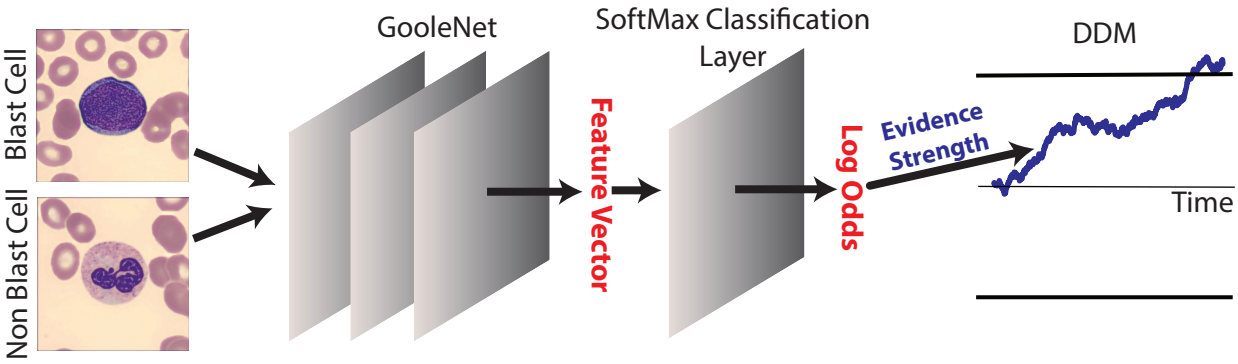

Figure A1. **Model Schematic:** A pre-trained GoogLeNet was augmented in order to use "transfer learning" to train a deep CNN on the blast cell data set. The resulting network then outputs a classification (Blast or non-Blast) probability for each image, which is transformed into a log-odds value $LO = \log(P(B)/P(NB))$. The drift rate for the subsequent diffusion model is then a linear function of this LO, $v = v_{inter} + v_{slope} * LO$.

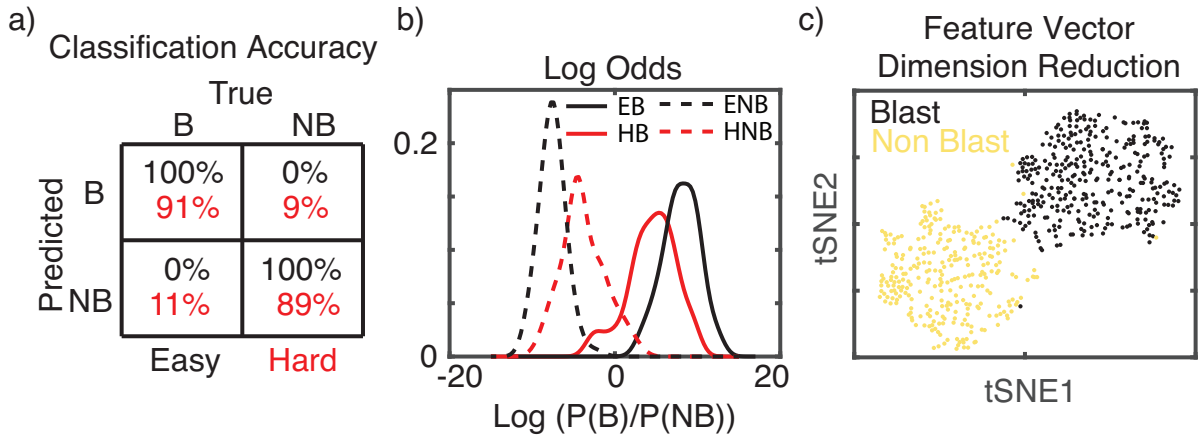

Figure A2. **CNN Analysis:** *Panel a)* Table showing the classification accuracy of the CNN for both easy and hard images. Note that the easy and hard designations were never utilized during training of the CNN and are only used for post-hoc analysis. *Panel b)* Distribution of log odds (LO) values for the four image classes. Results demonstrate that, at the distributional level, the network outputs smaller LO values (values closer to 0 that is) for images deemed to be harder to classify by experts. *Panel c)* tSNE plot showing a low dimensional representation of the feature vectors (FV) output by the CNN prior to classification. Colors indicating cell type are added after the tSNE application. Thus results show a natural clustering of the FVs into two clusters that corresponds to Blast and non-Blast cells.

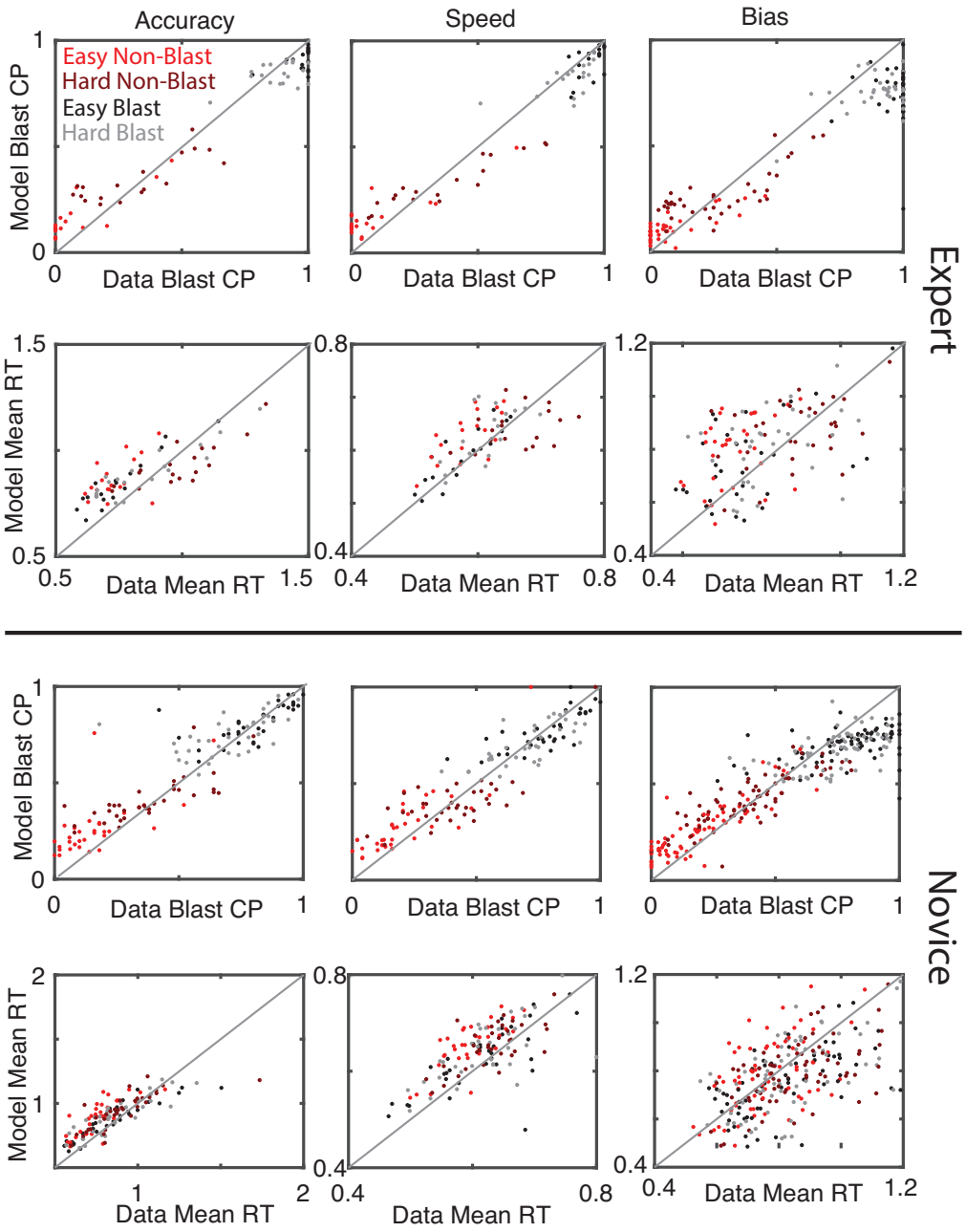

Figure A3. **CNN-DDM Quality of Model Fit:** Comparison of model predictions (vertical axes) and observed (horizontal axes) for response proportions (a-c for experts and g-i for novices) and mean response times (d-f for experts and j-l for novices) in the three instruction conditions. The solid diagonal line indicates perfect agreement where predictions and observations exactly coincide.

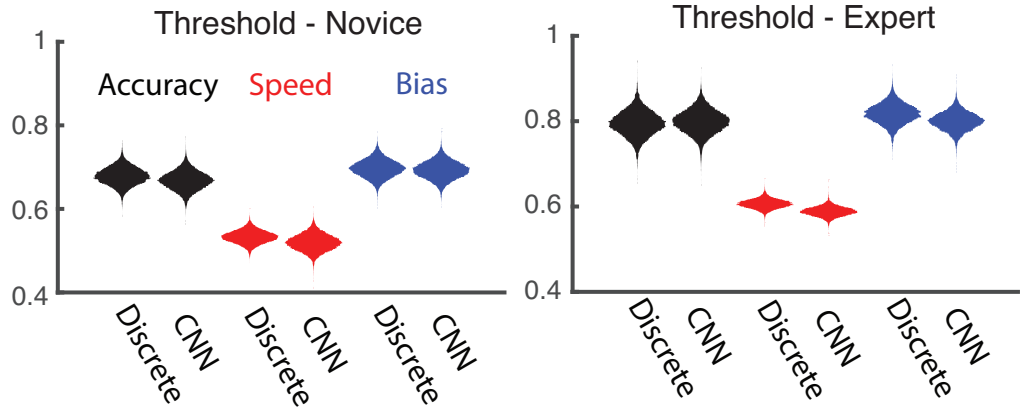

Figure A4. **Threshold Posteriors:** Violin plots showing the posterior distributions of the threshold values in the three experimental conditions for both novices and experts. Posteriors are grouped so that the thresholds estimated using the discrete difficulty and CNN based models can be directly compared.

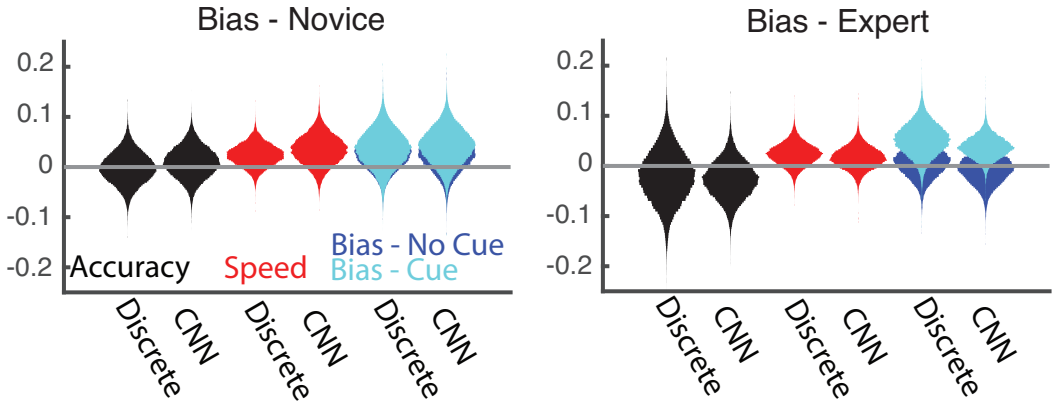

Figure A5. **Bias Posteriors:** Violin plots showing the posterior distributions of the bias values in the three experimental conditions for both novices and experts. Bias is quantified as a fraction of the threshold. Thus a bias of 0 indicates no bias and a bias of 0.05 indicates a start point bias that is 5% of the distance to the threshold. In the "Bias" conditions, separate bias parameters were estimated for the trials with and without the presentation of a cue (depicted with different shades of blue).

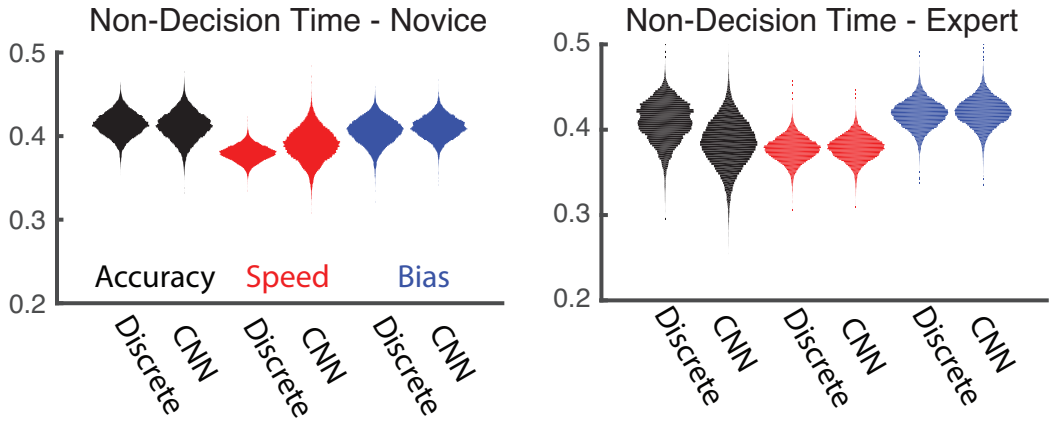

Figure A6. **Posterior Distributions of the Non Decision Time Parameter :** Violin plots showing the posterior distributions for the non decision time parameter for both models in fits of the speed, accuracy, and bias conditions.

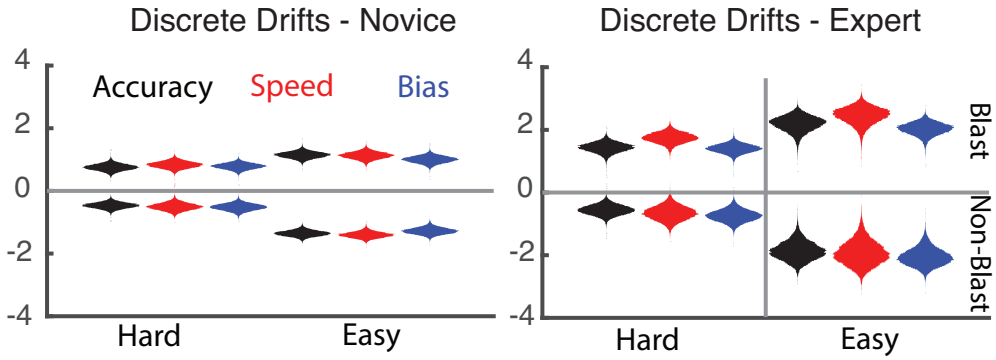

Figure A7. **Drift Rate Posteriors for the Discrete Difficulty Model:** Violin plots showing the posterior distributions of the estimated drift rate values in the three experimental conditions for both novices and experts. Separate drift rates were estimated for hard and easy image classes. Since there is no direct analogue of these drift rate parameters in the CNN based model, no comparison is made here. See Figure A8 for posterior estimates for the regression parameters that determine drift rate parameters in that model.

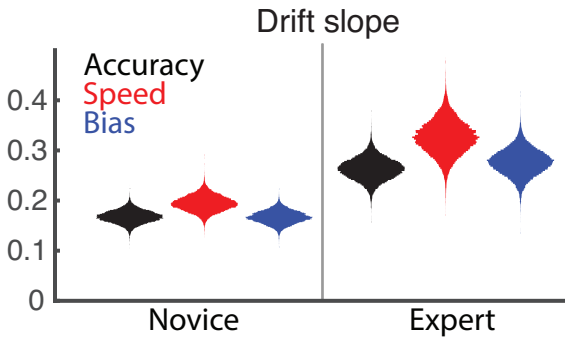

Figure A8. **Posterior Distributions of Drift Rate Slope Parameter in the CNN Based Model:** Violin plots showing the posterior distributions for the v_{slope} parameter. Recall that in the CNN based model, the drift rate associated with each image is determined by $v = v_{inter} + v_{slope} * LO$ where LO indicates the log odds value. Thus a higher value of v_{slope} indicates an increased ability to translate visual information encoded in the images into evidence for either alternative.