RTNet: A neural network that exhibits the signatures of human perceptual decision making

Farshad Rafiei*, Medha Shekhar* & Dobromir Rahnev

School of Psychology, Georgia Institute of Technology, Atlanta, GA

Keywords: Deep neural networks, reaction time, perceptual decision making, sequential sampling, confidence

Acknowledgments

This work was supported by the National Institute of Health (award: R01MH119189) and Office of Naval Research (award: N00014-20-1-2622). We thank Sashank Varma and Paul Verhaeghen for helpful suggestions about the analyses, as well as Ana Shin and Himanaga Sahithi Pandi for assistance with data collection.

Competing interests:

None

Correspondence

Farshad Rafiei
Georgia Institute of Technology
654 Cherry Str. NW
Atlanta, GA 30332
E-mail: farshadrafiei3@gmail.com

Abstract

Convolutional neural networks currently provide the best models of biological vision. However, their decision behavior, including the facts that they are deterministic and use equal number of computations for easy and difficult stimuli, differs markedly from human decision-making, thus limiting their applicability as models of human perceptual behavior. Here we develop a new neural network, RTNet, that generates stochastic decisions and human-like response time (RT) distributions, and also reproduces all foundational features of human accuracy, RT, and confidence. To test RTNet's ability to predict human behavior on novel images, we collected accuracy, RT, and confidence data from 60 human subjects performing a digit discrimination task. We found that the accuracy, RT, and confidence produced by RTNet for individual novel images correlated with the same quantities produced by human subjects. Critically, human subjects who were more similar to the average human performance were also found to be closer to RTNet's predictions. Overall, RTNet is a promising model of human response times that exhibits the critical signatures of perceptual decision making.

Introduction

Traditional cognitive models of perceptual decisions[1,2] are able to account for the major features of human perceptual decision making, but do not operate on the level of images. Recently, convolutional neural networks (CNNs) have reached and sometimes exceeded human-level performance for novel images[3,4]. In addition, these networks naturally handle multi-choice categorization tasks and currently provide the best models of the processing related to object recognition in the ventral visual stream of the human brain[3,5,6]. However, traditional CNNs' decision behavior differs markedly from human decision behavior, thus limiting their applicability as models of human perceptual decision making. Specifically, unlike humans, traditional CNNs are both deterministic (i.e., they always give the same response for a given stimulus) and static (i.e., they are invariant in the amount of time spent on processing different images and thus always produce the same response time) (**Figure 1A**).

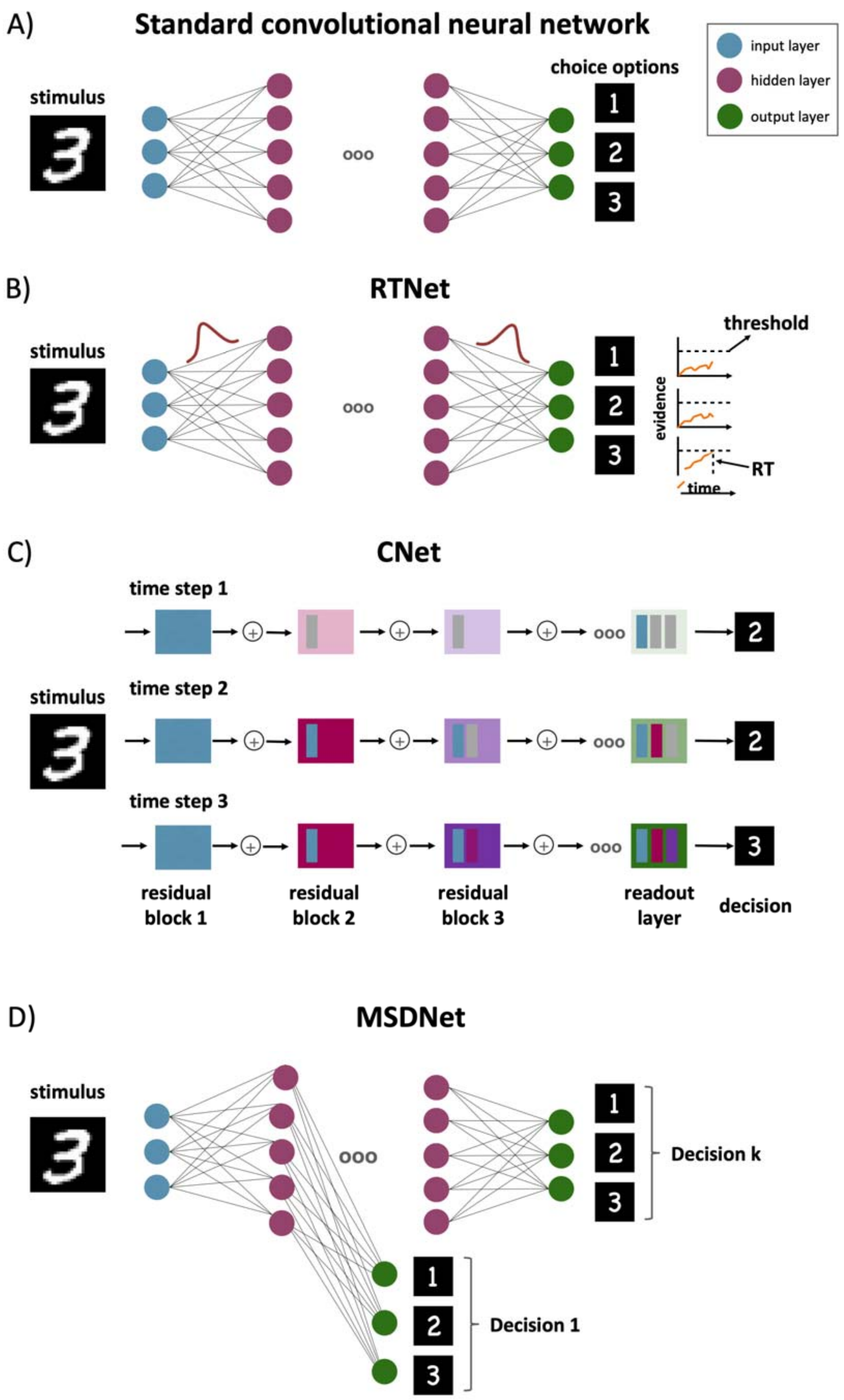
A)
Standard convolutional neural network
input layer
hidden layer
output layer
stimulus
3
ooo
choice options
1
2
3
B)
RTNet
stimulus
3
ooo
1
2
3
threshold
evidence
RT
time
C)
CNet
time step 1
ooo
2
stimulus
3
time step 2
ooo
2
time step 3
ooo
3
residual block 1
residual block 2
residual block 3
readout layer
decision
D)
MSDNet
stimulus
3
ooo
1
2
3
Decision k
1
2
3
Decision 1

Figure 1. Model architectures. (A) A standard feedforward CNN architecture that consists of an input layer, several hidden layers, and an output layer. All images receive the same amount of processing and therefore the network cannot account for variable RT. Because all weights are fixed, the network is deterministic (i.e., it always arrives at the same response for a given stimulus). (B) RTNet architecture. Unlike standard CNNs, the connection weights in RTNet are not fixed but are instead each chosen from a distribution. A stimulus is processed multiple times by the network, each time using a different set of randomly chosen weights. The evidence from each processing step is accumulated and a decision is made when the evidence for one of the choice options reaches a predefined threshold. This architecture results in both stochastic decisions and variable RT. (C) Parallel cascaded network (CNet) architecture. CNet utilizes skip connections to introduce propagation delays between residual blocks (each of which consist of two convolutional layers). At each time step, all residual blocks receive inputs from lower blocks and actively participate in computations. However, due to propagation delays between blocks, earlier blocks achieve stable activations faster, whereas the later blocks only receive partial updates from earlier blocks and therefore require multiple processing steps to achieve stable activations. For instance, residual block 1 (blue) achieves stable activations at time step 1, whereas residual block 3 (purple) requires three time steps to receive complete input from all the blocks below and achieve stable output. At any given processing step, the network can generate a decision via the readout layer, although if time step is less than the number of residual blocks, the decision will be based on partial input in later blocks (D) Multi-scale dense network (MSDNet) architecture. Similar to a standard feedforward CNN, MSDNet has a single input layer and several hidden layers. However, in this network, each hidden layer features its own classifier (i.e., its own output layer) allowing MSDNet to make a separate decision after the processing in each layer is completed. This allows the network to stop processing an image early if that image can already be decoded from earlier layers of the network, thus resulting in different RTs for different images (though a given image still always produces the same response and RT).

Several lines of work have tried to build mechanisms into neural networks to make them stochastic and dynamic[7–11]. Early research on shallow multi-layer perceptron (MLP) models was able to create models that were both stochastic and dynamic, and were able to explain human behavior on simple cognitive tasks[12–14]. However, these models are not image-computable (i.e., they cannot handle complex input such as images). More recent work has produced image-computable dynamic networks capable of generating response times (RTs). In these networks, the computational resources utilized for the decision increase with time[7–9], allowing responses to evolve through each processing step. However, although these networks can mimic the speed-accuracy tradeoff (SAT) found in humans, they are deterministic and their internal mechanisms are not well supported by existing models of human perception and cognition.

Here we combine modern CNNs with traditional cognitive models to create a model that is image-computable, stochastic and dynamic, and can reproduce the critical features of perceptual decision making for novel images. The model, which we call RTNet for its ability to model human RTs, features a deep convolutional neural network with noisy weights and processes a given image several times using a different random sample of these weights in each processing step (**Figure *1*B**). These weights are sampled from a Bayesian neural network (BNN) that estimates a posterior distribution over the best network parameters learnt during training. By varying the weights from one processing step to another, the network's units produce variable responses to the same input that mimic the randomness of neural responses. After each processing step, RTNet accumulates the output corresponding to each choice until one of the choices reaches a predefined threshold. The model therefore has a strong conceptual relationship to race models from the cognitive literature on decision-making, which postulate a noisy accumulation process with separate accumulators for each choice[15–17]. By combining the image-computability of CNNs with traditional models of perception, we expect RTNet to be applicable across a wide range of perceptual tasks as well as reproduce the basic features of human perceptual decision making.

To assess a model's ability to make decisions similar to humans, one needs to test whether it produces the foundational features of human decision-making[18]. Human perceptual decision making has been studied primarily in the context of 2-choice tasks using artificial stimuli such as Gabor patches or random dot motion[19] (although notable exceptions exist where N-choice tasks are used[20–23]). Therefore, we first replicate the known decision-making signatures from such tasks using an 8-choice task with meaningful images (hand-written digits taken from the MNIST dataset[24]). We manipulate 1) task difficulty by adding two different levels of noise to the images, and 2) speed-accuracy trade-off (SAT) by asking subjects to emphasize either the accuracy or speed of their responses on different trials.

Critically, we test RTNet under the same conditions and with the same images seen by the human subjects to explore the model's capability to produce behavior similar to human agents. Beyond testing whether RTNet can reproduce the basic features of human perceptual decision making, we also explore whether the accuracy, RT, and confidence produced by RTNet for individual images predict the corresponding quantities for humans on the same images. Finally, throughout the paper, we compare the behavior of RTNet to that of two other popular dynamic CNN models – Parallel Cascaded Networks[7] (CNet; **Figure 1C**), which is currently thought to be the best image-computable model of human RT[10], and Multi-Scale Dense Networks[11] (MSDNet; **Figure *1*D**), which implements one of the most common ways for generating RTs in image-computable models. We find that RTNet's behavior mimics human perceptual decision making better than both of these models.

Results

We collected data from 60 human subjects who performed a digit discrimination task (**Figure 2A**). The experiment was a 2 x 2 design with factors of task difficulty (easy vs. difficult images) and speed pressure (speed vs. accuracy focus). Each condition consisted of 120 unique images, and each subject made a decision regarding each image exactly twice, which allowed us to determine the level of stochasticity in human behavior (**Figure 2B**). Overall, each subject completed 960 trials in total.

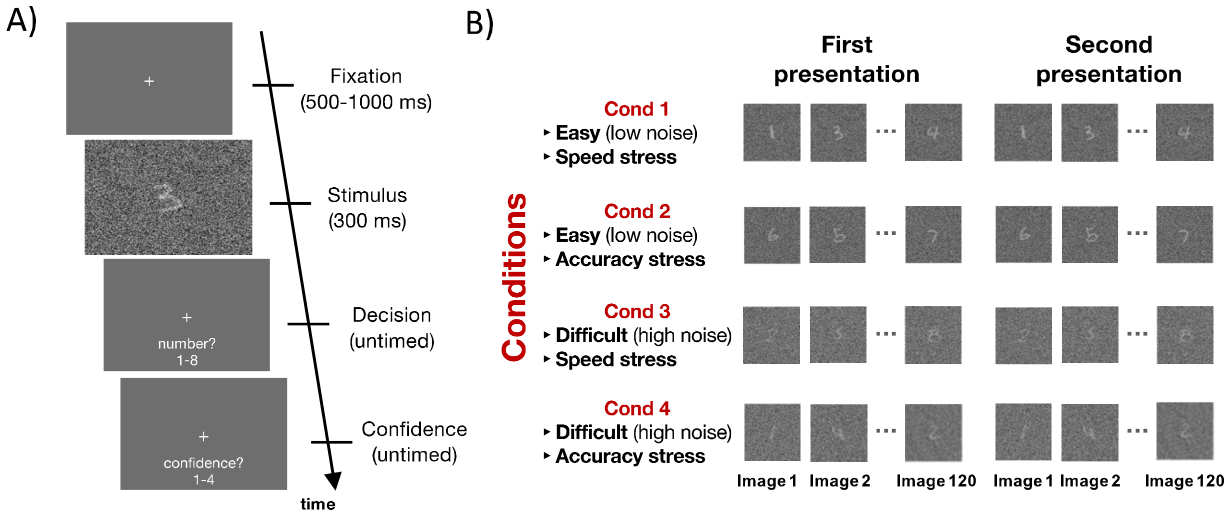

Figure 2. Experiment task. (A) Trial structure. Each trial began with a fixation cross presented for 500 to 1000 ms, followed by an image of a hand-written digit from the MNIST dataset embedded in noise and presented for 300 ms. Only the digits 1-8 were used. Subjects reported their choice and confidence (on a 4-point scale) using separate, untimed button presses. Note that the noisy stimulus subtended a visual angle of 6.06° and did not cover the entire screen. (B) Experimental design. The experiment included four conditions such that subjects judged easy (low noise) or difficult (high noise) images while emphasizing either speed or accuracy. Each condition featured 120 unique images that were the same across all subjects (total of 480 unique images in the experiment). In addition, each image was presented twice to allow the estimation of the stochasticity of human perceptual choices. Each subject thus completed a total of 960 trials. The images within the first and second sets of presentation were shown in a different random order.

Having obtained these human data, we compared the human behavior to that of RTNet, CNet and MSDNet. Both RTNet and MSDNet were implemented using the eight-layer AlexNet architecture with five convolutional layers followed by three fully connected layers[25]. CNet was based on the architecture of ResNet18, since the implementation of this model relies on residual blocks and skip connections. Given that humans and deep learning models are impacted differently by stimulus noise[26,27], we adjusted the noise levels of the images seen by each network to match their overall accuracy to the accuracy produced by the human subjects. In addition, to allow the networks to reproduce the speed-accuracy trade-off observed in the human data, we adjusted the threshold value that triggers a decision for each model as to match the human accuracy separately in the speed- and accuracy-focused conditions. To improve the correspondence between the model predictions and the human data, we trained 60 instances of each model (by only changing the initial parameters before training began) and analyzed the data produced by these 60 instances in equivalent manner to the 60 human subjects.

Signatures of human decision-making

We examined six foundational signatures of human perceptual decision making that have already been established in studies of 2-choice tasks: 1) Human decisions are stochastic, meaning that the same stimulus can elicit different responses on different trials[28,29], 2) increasing speed stress shortens RT but decreases accuracy (speed-accuracy trade-off)[18,30,31], 3) more difficult decisions lead to reduced accuracy and longer RT[18,32,33], 4) RT distributions are right-skewed, and this skew increases with task difficulty[18], 5) RT is lower for correct than for error trials[33–37], and 6) confidence is higher for correct than for error trials[38]. For each of these signatures, we confirmed that the signature also occurs for our 8-choice task with naturalistic images, and then tested whether RTNet, CNet and MSDNet exhibit the same signature.

Stochasticity of human decisions

A central feature of human behavior is that human decisions are stochastic such that the same stimulus can elicit different responses on different trials[28,29,39]. We quantified the level of stochasticity in each condition by presenting each image twice. On average across all conditions, 36% of all images received different responses on the two presentations (one-sample t-test: t(59) = 36.78, $p < 0.0001$) (**Figure *3*A**). A repeated measures ANOVA with factors stimulus difficulty (easy vs. difficult) and SAT (speed vs. accuracy stress) revealed that stochasticity increased with both higher task difficulty (F(1,63) = 871.87, $p < 0.0001$) and higher speed pressure (F(1,63) = 9.14, $p = 0.0036$).

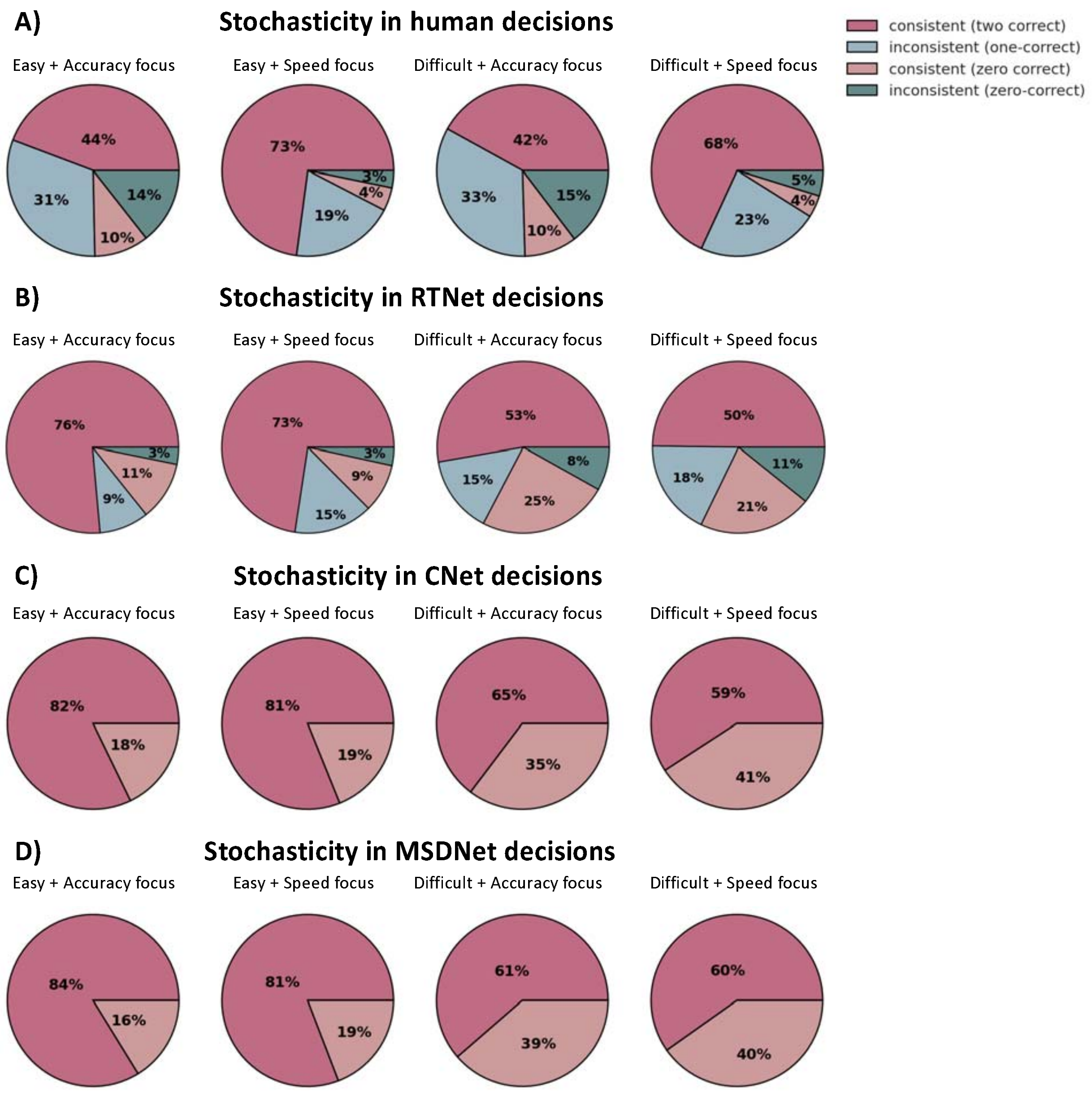

Figure 3. Decision stochasticity in humans and all networks. Stochasticity of decisions made by (A) humans, (B) RTNet, and (C) CNet and (D) MSDNet. Warm colors indicate that the same response was given both times an image was presented (whether the response was correct or incorrect), whereas cool colors indicate that different responses were given for the two image presentations (whether or not any of them was correct). Humans and RTNet exhibit stochastic decision-making with stochasticity increasing with task difficulty and speed stress. However, CNet and MSDNet in their standard versions are fully deterministic. In the legend, "consistent (two correct)" refers to instances when the correct responses was given for both presentations of a given image, "consistent (zero correct)" refers to instances when the same incorrect choice was made both times, "inconsistent (one correct)" refers to instances when only one of the

choices was correct, “inconsistent (zero correct)” refers to instances where different incorrect choices were made each time.

Due to the fact that RTNet uses a random sample of weights for each processing step, it naturally produces stochastic decisions too. On average across all conditions, RTNet produced different responses on the two image presentations on 20% of trials (t(59) = 32.65, $p < 0.0001$; **Figure *3*B**). This level of stochasticity was lower than for human subjects and stems from the fact that the variability in the weights was fixed a priori by training a Bayesian neural network. However, increasing the variability of the weights can increase the stochasticity of the decisions made by RTNet. Further, the stochasticity in human decisions partially stems from factors such as fluctuations in attention, arousal, or serial dependence[28,29,39,40], which we did not attempt to model. Because of these considerations, we did not try to match RTNet to the exact level of human decision stochasticity observed in the data. Critically, however, RTNet exhibited the same features such that stochasticity increased with higher task difficulty (F(1,59) = 120.12, $p < 0.0001$) and higher speed stress (F(1,59) = 87.73, $p < 0.0001$). On the other hand, for a fixed level of speed-accuracy trade-off, both CNet and MSDNet are fully deterministic and should not exhibit any decision stochasticity, which we confirmed in our simulations (**Figure *3*C,D**). We note that it should be possible to add noise in the weights of these models to induce stochastic decisions, but such noise would decrease their accuracy much more than it affects RTNet given that only RTNet is able to average out the noise over repeated processing steps.

Speed-accuracy trade-off

The ability to trade off speed and accuracy against each other is a hallmark of decision-making across humans and many other animal species[30,31]. The human data confirmed that increased speed pressure led to lower accuracy (F(1,59) = 4.27, *p* = 0.0431; **Figure *4*A**) and shorter RTs (F(1,59) = 119.29, *p* < 0.0001; **Figure *4*B**). All models were able to replicate this pattern. Specifically, increased speed pressure resulted in lower accuracy for RTNet (F(1,59) = 9.68, *p* = 0.0029), CNet (F(1,59) = 50.03 , *p* < 0.0001), and MSDNet (F(1,59) = 21.84 , *p* < 0.0001). Increased speed pressure also led to shorter RTs for RTNet (F(1,59) = 3362.57, *p* < 0.0001), CNet (F(1,59) = 695.88, *p* < 0.0001), and MSDNet (F(1,59) = 584.08, *p* < 0.0001). These results indicate that speed-accuracy trade-off is robustly observed even for relatively complex task with naturalistic images, and that all three models examined here exhibit this foundational phenomenon.

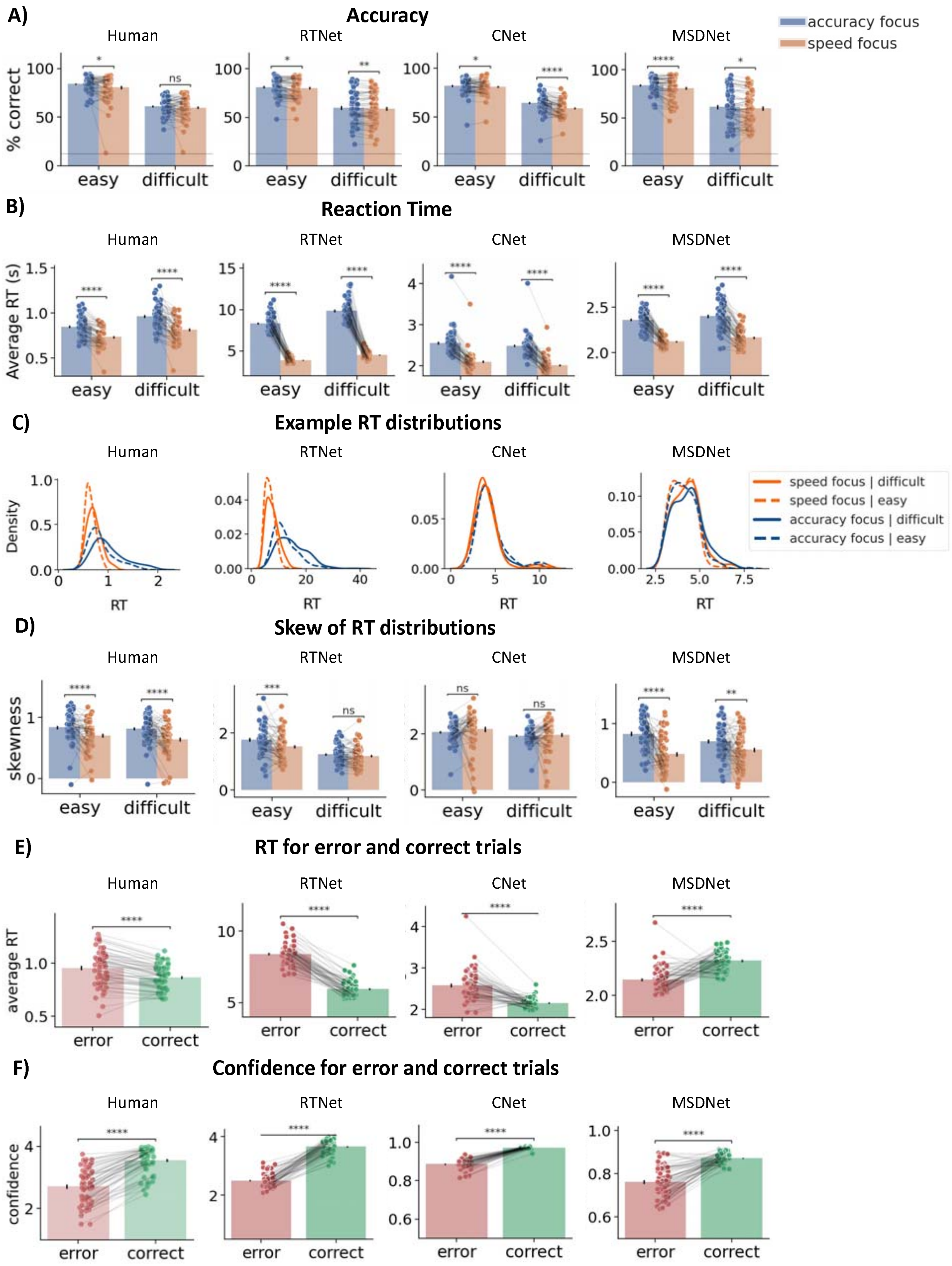
A)
Accuracy
Human
RTNet
CNet
MSDNet
accuracy focus
speed focus
% correct
0
50
100
easy
difficult
*
ns
**
****
B)
Reaction Time
Average RT (s)
easy
difficult
****
C)
Example RT distributions
Density
RT
speed focus | difficult
speed focus | easy
accuracy focus | difficult
accuracy focus | easy
D)
Skew of RT distributions
skewness
***
ns
**
E)
RT for error and correct trials
average RT
error
correct
F)
Confidence for error and correct trials
confidence
error
correct

Figure 4. Behavioral effects shown by human subjects and the models. (A) Accuracy for humans decreases when response speed is emphasized as well as for more difficult decisions. Both effects are exhibited by all the networks. (B) RT for humans becomes shorter when response speed is emphasized, as well as for easier decisions. Both effects are also exhibited robustly by RTNet. However, while both CNet and MSDNet produced a robust effect for the speed manipulation, they exhibited much smaller effects for the difficulty manipulation. (C) RT distributions for a representative subject/model. (D) The skewness of RT distributions change across conditions. For humans and RTNet, the skewness of the RT distributions was higher for easier tasks and for accuracy focus. However, while CNet showed the same effect for the difficulty manipulation, it failed to demonstrate skewness differences for the SAT manipulation. On the other hand, MSDNet showed the same effect for accuracy focus, but failed to exhibit skewness differences between easy and difficult decisions. (E) For humans, RTNet and CNet, error trials were associated with higher RT than correct trials. However, MSDNet showed the opposite pattern such that correct trials were associated with longer processing time. (F) Confidence for correct trials was higher than confidence for error trials for humans and all networks. For all panels, dots represent individual subjects; error bars show SEM; *p<0.05; **p<0.01; ***p<0.001, ****p<0.0001; n.s., not significant.

More difficult decisions lead to reduced accuracy and longer RT

Another ubiquitous feature of decision-making is that more difficult stimuli lead to lower accuracy and longer RT[18,41]. Our human data robustly showed this effect with more difficult stimuli leading to lower accuracy (F(1,59) = 1558.50, $p < 0.0001$; **Figure *4*A**) and longer RT (F(1,59) = 411.15, $p < 0.0001$; **Figure *4*B**). The same pattern was robustly observed for RTNet, where difficult stimuli led to lower accuracy (F(1,59) = 218.51, $p < 0.0001$) but longer RT (F(1,59) = 223.45, $p < 0.0001$). However, while CNet and MSDNet also showed a very robust effect on accuracy (CNet: F(1,59) = 1116.80, $p < 0.0001$; MSDNet: F(1,59) = 247.52, $p < 0.0001$), they exhibited a smaller effect for RT (CNet: F(1,59) = 6.17, $p = 0.0158$; MSDNet: F(1,59) = 11.07, $p = 0.0015$). Indeed, out of the 60 model instances, only 23 CNet instances and 36 MSDNet instances exhibited an RT increase for more difficult stimuli, while this effect was present in 60/60 human subjects and 58/60 RTNet instances. These results indicate that the effect of task difficulty on accuracy is exhibited robustly in humans and all networks, but the effect of task difficulty on RT is larger for humans and RTNet compared to CNet and MSDNet.

Skewness of RT distributions

For simple 2-choice decisions, human RT distributions are generally positively skewed and the skewness changes as a function of task conditions[2,18]. Our 8-choice task produced RT distributions that closely resemble what is observed in standard 2-choice tasks (**Figure *4*C**). Similar-looking RT distributions were produced by RTNet but MSDNet produced RT distributions that, while still right-skewed, exhibited a much sharper drop-off after their peak (**Figure *4*C**). We further assessed how the skewness of the RT distributions changed under different conditions. We found higher skewness for accuracy compared to speed focus (F(1,59) = 32.84, $p < 0.0001$), as well as for easy compared to difficult stimuli (F(1,59) = 5.10, $p = 0.0277$; **Figure *4*D**). RTNet exhibited the same pattern with skewness increasing with a focus on accuracy (F(1,59) = 15.32, p = 0.0002) and with easier stimuli (F(1,59) = 84.50, p < 0.0001). For CNet, we found no difference in skewness of RT distributions between the SAT conditions (F(1,59) = 1.05, p = 0.31), but skewness increased for easy compared to difficult stimuli (F(1,59) = 8.02, p = 0.006). On the other hand, while MSDNet showed an increase in skewness with a focus on accuracy (F(1,59) = 52.75, $p < 0.0001$), it produced RT distributions that did not significantly differ in skewness between the task difficulty conditions (F(1,59) = 0.72, $p = 0.40$). Overall, RTNet produced RT distributions which reflected the observed patterns in human data better than both Cnet and MSDNet. It should be noted that Cnet and MSDNet can only produce distinct RTs that are less than or equal to its layer numbers, which may affect their ability to reproduce human RT distributions unless a relatively high number of layers is used. On the other hand, RTNet is capable of going through arbitrary number of samples regardless of the number of layers in its architecture.

RT is faster for correct compared to error trials

Another ubiquitous feature of human behavior in 2-choice tasks is that correct decisions are typically accompanied by faster RTs than incorrect decisions[33–37]. We replicated this effect in our 8-choice task (F(1,59) = 82.08, $p < 0.0001$; **Figure *4*E**). The same difference between correct and error RTs also emerged for RTNet (F(1,59) = 831.15, $p < 0.0001$) and Cnet (F(1,59) = 83.92, $p < 0.0001$). However, MSDNet exhibited the opposite pattern such that RTs were faster for error compared to correct trials (F(1,59) = 65.70, $p < 0.0001$). This behavior is due to the fact that errors produced by MSDNet come mostly from decisions made in earlier layers. It may be possible to reverse this behavior by using a much more conservative decision threshold in the early compared to the late layers of MSDNet, though the effectiveness of this strategy and its effect on all other behavioral signatures examined here would need to be tested. What is clear is that MSDNet in its current form makes a qualitatively wrong prediction regarding the difference between correct and error RT, whereas RTNet and Cnet naturally reproduce the empirical effect.

Confidence is higher for correct than error trials

Finally, a ubiquitous feature of confidence ratings is that they are higher for correct compared to incorrect decisions[38,42]. Our human data replicated this effect (F(1,59) = 472.17, $p < 0.0001$;

Figure *4*F). The effect was also robustly exhibited by all three networks: RTNet (F(1,59) = 966.80, $p < 0.0001$), Cnet (F(1,59) = 785.99, $p < 0.0001$) and MSDNet (F(1,59) = 131.92, $p < 0.0001$). Therefore, humans and all networks robustly showed higher confidence for correct trials compared to incorrect trials.

Model predictions for accuracy, RT, and confidence for individual images

The results above demonstrate that RTNet is able to reproduce all foundational features of human decision-making. On the other hand, both CNet and MSDNet fail to exhibit stochastic decisions and skewness difference in RT distributions between the SAT/difficulty conditions, and MSDNet further fails to account for lower RT for correct decisions. However, RTNet's ability in those respects can easily be matched by traditional cognitive models that do not work on image-level data[16,34,43]. Therefore, a critical advantage of RTNet over traditional cognitive models would be the ability to predict human behavior for individual, unseen images because traditional models cannot do that. Here we tested specifically whether the accuracy, RT, and confidence for unseen images produced by RTNet, CNet and MSDNet predict the same quantities in humans.

Model predictions across all conditions for individual subjects

In a first set of analyses, we assessed the correlations between the accuracy, RT, and confidence for each human subject and the corresponding quantities predicted by RTNet, CNet and MSDNet across all four conditions (easy with speed stress, difficult with speed stress, easy with accuracy stress, difficult with accuracy stress). We compared how well data from individual human subjects could be predicted by RTNet, CNet, MSDNet, as well as from the data from the 59 remaining human subjects. This last quantity, which we call subject-to-group relationship, provides an estimate of the noise ceiling (i.e., the performance that a true model could achieve given inter-subject variability)[44].

We found that all models predicted individual human data much better than chance for accuracy, RT and confidence (all *p*'s < 0.0001). However, RTNet provided substantially better predictions than both other models (**Figure *5***). This was true for accuracy (Difference with CNet: t(59) = 18.63, *p* < 0.0001; Difference with MSDNet: t(59) = 13.31, *p* < 0.0001), RT (Difference with CNet: t(59) = 30.67, *p* < 0.0001; Difference with MSDNet: t(59) = 30.67, *p* < 0.0001), and confidence (Difference with CNet: t(59) = 8.39, *p* < 0.0001; Difference with MSDNet: t(59) = 7.68, *p* < 0.0001).

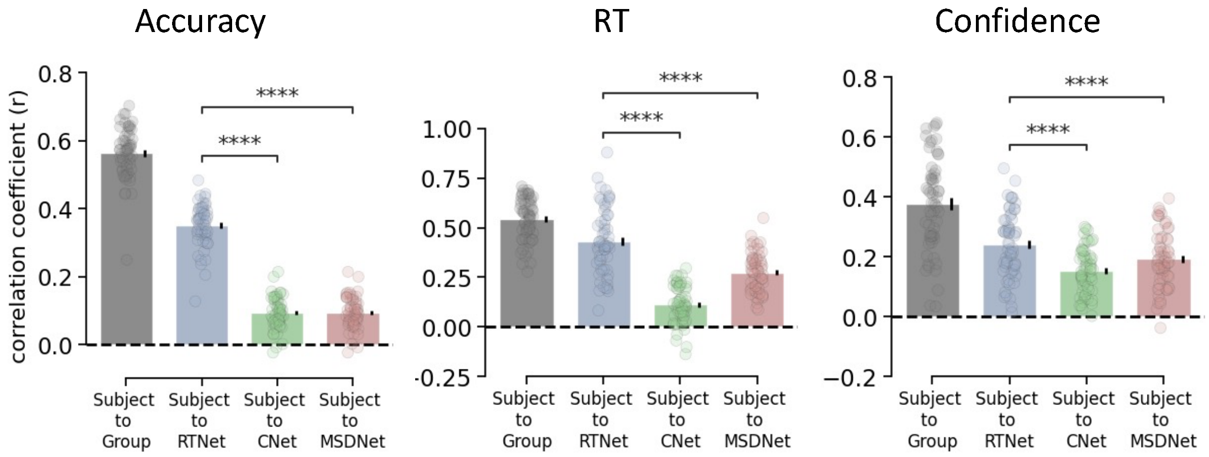

Figure 5. Image-by-image correlation between human data and each model across all experimental conditions for individual subjects. Correlation between data from individual human subjects and the group average/RTNet/CNet/MSDNet for accuracy, RT, and confidence across all conditions. The strength of correlation is stronger for RTNet than CNet or MSDNet for each measure. The subject-to-group correlation provides an estimate of the noise ceiling for the correlations. Dots represent individual subjects; error bars show SEM; ****p<0.0001.

Critically, RTNet's predictions were reasonably close to the noise ceiling in all cases (calculated as the average subject-to-group correlation in the human data). Specifically, RTNet's predictions were within 62.5%, 79.6%, and 64.8% of the noise ceiling for accuracy, RT and confidence, respectively. These numbers were substantially lower for CNet (16.1%, 20.3%, 40.5%, respectively) and MSDNet (16.1%, 50%, and 51.3%, respectively). Thus, by reaching to between 62.5% and 79.6% of the noise ceiling, RTNet can provide excellent predictions for the accuracy, RT, and confidence produced by human subjects for images that the model was not trained on. Additionally, we derived the model predictions for averages across the 60 subjects across all conditions (**Supplementary Figure 1**) and found that RTNet predicts average human accuracy and RT better than the other networks.

Model predictions within each condition separately

The analyses above explored the correlations between model predictions and human behavior across all experimental conditions. Because different conditions vary in their average accuracy, RT, and confidence, analyses across conditions are likely to produce higher correlations than if the same analyses are to be performed within each condition separately. Therefore, we repeated the analyses above but within each of the four conditions separately to investigate if the two models can still account for accuracy, RT, and confidence on individual images. We found that RTNet and MSDNet produced accuracy, RT, and confidence predictions that significantly correlate with individual subject data in all conditions (all *p*'s < 0.0001; **Figure 6**).

However, for the RT predictions produced by CNet for all conditions except speed focus with easy images, the correlations were either zero or negative (p's > 0.62).

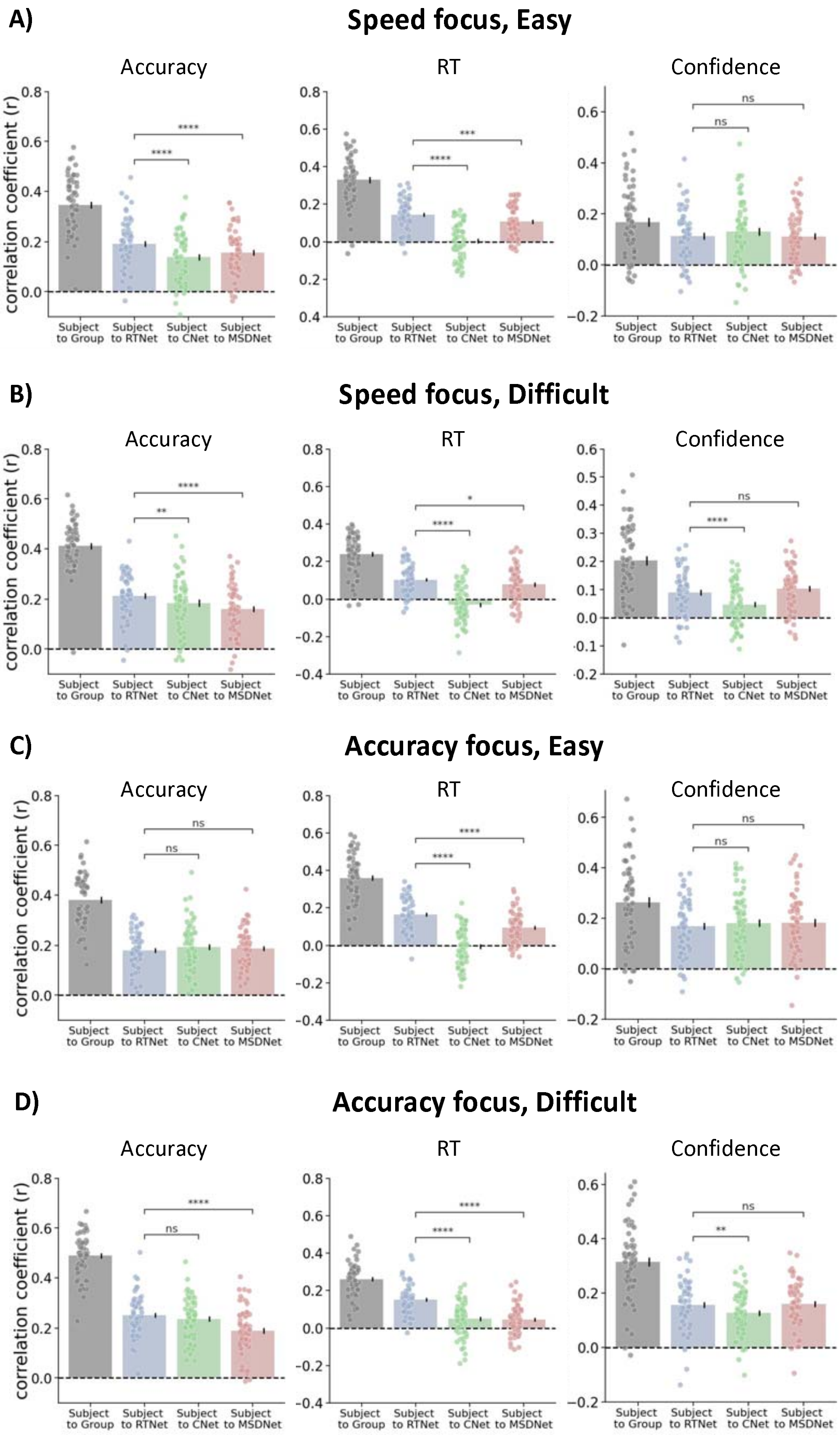
A)
Speed focus, Easy
Accuracy
RT
Confidence
correlation coefficient (r)
0.8
0.6
0.4
0.2
0.0
-0.2
-0.4
****
****
***
ns
ns
Subject to Group
Subject to RTNet
Subject to CNet
Subject to MSDNet
B)
Speed focus, Difficult
Accuracy
RT
Confidence
correlation coefficient (r)
0.5
0.3
0.1
-0.1
****
**
*
****
ns
C)
Accuracy focus, Easy
Accuracy
RT
Confidence
correlation coefficient (r)
ns
ns
****
****
ns
ns
D)
Accuracy focus, Difficult
Accuracy
RT
Confidence
correlation coefficient (r)
****
ns
****
****
ns
**

Figure 6. Image-by-image correlation between human data and each within each experimental condition. Correlation between data from individual human subjects and the group average/RTNet/CNet/MSDNet for accuracy, RT, and confidence within each experimental condition – A) speed focus; easy, B) speed focus; difficult, C) accuracy focus; easy, D) accuracy focus; difficult. The strength of correlation is significantly stronger for RTNet than CNet in eight out of the 12 comparisons and seven out of 12 comparisons for MSDNet; RTNet never exhibits significantly weaker correlations than either CNet or MSDNet. For all panels, dots represent individual subjects; error bars show SEM; *p<0.05; **p<0.01; ***p<0.001, ****p<0.0001; n.s., not significant.

Critically, however, RTNet predicted the individual data significantly better than CNet in two out of four conditions for accuracy (all *p*'s < 0.001), in all four conditions for RT (all *p*'s < 0.0001) and in two out of four conditions for confidence (*p* < 0.005). Compared to MSDNet, RTNet predicted the individual data significantly better in three out of four conditions for accuracy (all three *p*'s < 0.001) and in all four conditions for RT (all *p*'s < 0. 02). However, there was no significant difference in correlations for confidence predictions between RTNet and MSDNet for any of the confidence conditions (all *p*'s > 0.05). RTNet was never significantly worse than either CNet or MSDNet in predicting any of the 12 conditions. Overall, these results demonstrate that RTNet predicts human behavior well across all three measures and across different types of analyses (across- or within-condition), and does so better than CNet and MSDNet.

<u>Humans who are more similar to the group average are also more similar to RTNet</u>

Our subject-to-group analyses revealed substantial variability in how well individual subjects' data corresponded to the group average (see **Figure 5**). Since the group average constitutes the best model of human behavior, this variability indicates that different individuals deviate differently from the best model. Therefore, one would expect that the strength of the relationship for an individual subject and the group would be linked to the strength of the relationship of that same subject and any good model of behavior. Here we tested if such dependency holds true for RTNet, CNet and MSDNet. We found that subjects who exhibited greater correlation in image-by-image accuracy across all conditions with rest of the group also exhibited greater correlation with the RTNet predictions (r = 0.69, *p* < 0.0001; **Figure 7A**). The same correspondence also emerged for RT (r = 0.83, *p* < 0.0001) and confidence (r = 0.89, *p* < 0.0001). Similar results were obtained for CNet (Accuracy: r = 0.39, *p* < 0.0001; RT: r = 0.80, *p* < 0.0001; Confidence: r = 0.85, *p* < 0.0001; **Figure 7B**) and MSDNet (Accuracy: r = 0.39, *p* < 0.0001; RT: r = 0.43, *p* < 0.0001; Confidence: r = 0.64, *p* < 0.0001; **Figure 7C**), demonstrating that all three models predict better the data from individuals who behave more similarly to the rest of the group. Nevertheless, all correlations were highest for RTNet compared to CNet and MSDNet.

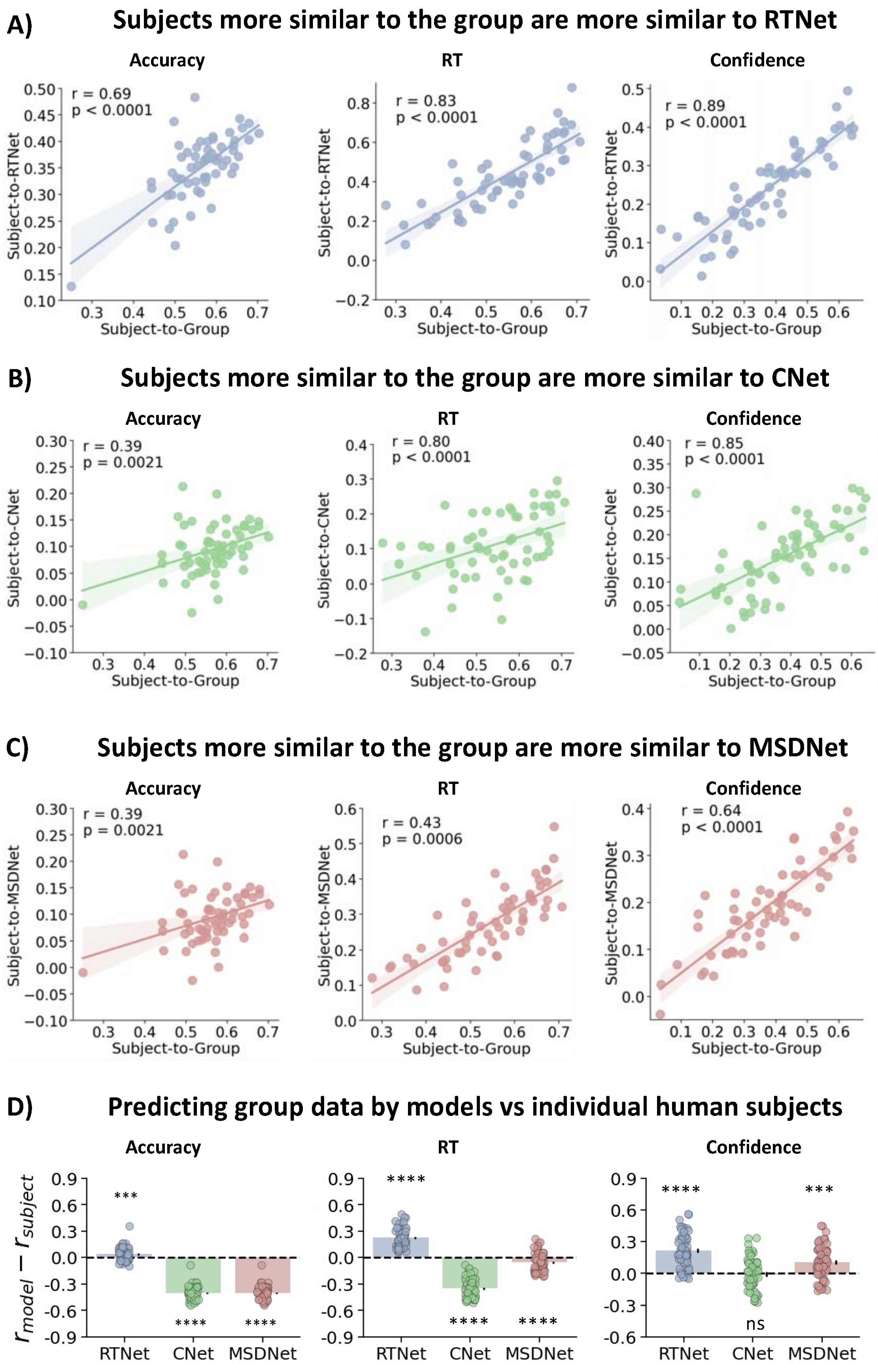
A)
Subjects more similar to the group are more similar to RTNet
Accuracy
r = 0.69
p < 0.0001
Subject-to-RTNet
0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50
0.3 0.4 0.5 0.6 0.7
Subject-to-Group
RT
r = 0.83
p < 0.0001
−0.2 0.0 0.2 0.4 0.6 0.8
0.3 0.4 0.5 0.6 0.7
Confidence
r = 0.89
p < 0.0001
0.0 0.1 0.2 0.3 0.4 0.5
0.1 0.2 0.3 0.4 0.5 0.6
B)
Subjects more similar to the group are more similar to CNet
Accuracy
r = 0.39
p = 0.0021
Subject-to-CNet
−0.10 −0.05 0.00 0.05 0.10 0.15 0.20 0.25 0.30
RT
r = 0.80
p < 0.0001
−0.2 −0.1 0.0 0.1 0.2 0.3 0.4
Confidence
r = 0.85
p < 0.0001
−0.05 0.00 0.05 0.10 0.15 0.20 0.25 0.30 0.35 0.40
C)
Subjects more similar to the group are more similar to MSDNet
Accuracy
r = 0.39
p = 0.0021
Subject-to-MSDNet
RT
r = 0.43
p = 0.0006
0.0 0.1 0.2 0.3 0.4 0.5 0.6
Confidence
r = 0.64
p < 0.0001
0.0 0.1 0.2 0.3 0.4
D)
Predicting group data by models vs individual human subjects
Accuracy
$r_{model} - r_{subject}$
0.9 0.6 0.3 0.0 -0.3 -0.6 -0.9
***
****
****
RTNet CNet MSDNet
RT
****
****
****
Confidence
0.9 0.6 0.3 0.0 -0.3 -0.6
****
ns
***

Figure 7. Humans who are more similar to the group average are also more similar to each model. (A) We observed a strong positive correlation between the subject-to-group and subject-to-RTNet similarity values for accuracy, RT, and confidence. This indicates that individual subjects whose behavior was more similar to the group average on per image basis were also more similar to the predictions made by RTNet. (B,C) Similar results were also observed for CNet and MSDNet, although these correlations tended to be lower than for RTNet. (D) Comparison between individual subjects and the models in predicting the group data. RTNet significantly outperformed individual human subjects in predicting group accuracy, RT, and confidence. On the other hand, CNet and MSDNet were worse than individual humans in predicting accuracy and RT, but MSDNet was better at predicting confidence. For all panels, dots represent individual subjects; error bars show SEM; *p<0.05; **p<0.01; ***p<0.001, ****p<0.0001; n.s., not significant.

Given the variability in how similar individual subjects were to the group data, we also explored how well RTNet, CNet and MSDNet compare to the ability of individual subjects to predict the group data. We found that RTNet outperformed individual human subjects in predicting the accuracy (t(59) = 4.08, *p* = 0.0001), RT (t(59) = 16.17, *p* < 0.0001), and confidence (t(59) = 10.92, *p* < 0.0001) of the rest of group across all conditions (**Figure 7D**). Impressively, RTNet outperformed every individual human subject in predicting the group RT and confidence results, as well as 73.3% of individual subjects in predicting accuracy. On the other hand, CNet and MSDNet performed significantly worse than individual humans in predicting accuracy (CNet: t(59) = -42.42, *p* < 0.0001; MSDNet: t(59) = -42.42, *p* < 0.0001) and RT (CNet: t(59) = -25.43, *p* < 0.0001; MSDNet: t(59) = -4.01, *p* = 0.0002) of the group. However, CNet was not significantly worse than individual humans in predicting confidence (t(59) = -0.36, *p* = 0.71), whereas MSDNet predicted the group confidence results better than most humans (t(59) = 5.26, *p* < 0.0001). In sum, RTNet outperformed most individual subjects in predicting the group data for accuracy, RT, and confidence, but this was not true for CNet or MSDNet.

Discussion

There is considerable interest in using neural networks as models of human visual processing and behavior, but relatively little work has been done on testing the extent to which existing image-computable models reproduce the full range of behavioral signatures exhibited by humans. Here we show that the current state-of-the-art neural networks such as Parallel Cascaded Networks (CNet) and Multi-scale Dense Networks (MSDNet) diverge in several ways from human behavior. Further, we develop a new neural network, RTNet, that exhibits all critical features of human perceptual decision making, including effects on accuracy, RT, and confidence. Further, RTNet predicted well human group behavior for novel images and did so better than both CNet and MSDNet, as well as better than individual human subjects. Finally, individual humans who were more similar to the group were also more similar to RTNet. Overall, RTNet provides the best current image-computable model of human accuracy, RT, and confidence.

Relationship between RTNet and cognitive models of perceptual decision making

RTNet is the first neural network to exhibit all critical signatures of human perceptual decision making. This success, however, is hardly surprising given the strong conceptual similarity between RTNet and traditional cognitive models of decision-making that also exhibit the signatures of human behavior[16,18,32,43,45]. These models are often referred to as sequential sampling models where (usually noisy) evidence is accumulated over time until a threshold is reached. The most common sequential sampling models are diffusion models, which are typically only applied to 2-choice tasks where evidence in favor of one response alternative is also evidence against the other alternative[1,32]. Instead, RTNet is conceptually more similar to another subgroup of sequential sampling models called race models where each choice option has its own accumulation system and evidence for each choice is accumulated in parallel[34,46].

Despite their conceptual similarity, RTNet has two important advantages over traditional cognitive models. Most importantly, RTNet is image-computable and can be applied to actual images, whereas traditional models cannot. As such, traditional models cannot replicate RTNet's ability to make accurate predictions regarding human accuracy, RT, and confidence for individual unseen images. The second advantage stems from the inability of traditional cognitive models to naturally capture the relationships between the different choice options. Specifically, to maintain a low number of free parameters, cognitive models are often fit with the assumption that evidence accumulates at the same rate for all incorrect choice options (but accumulates faster for the correct choice)[47]. However, this assumption ignores the fact that some incorrect options may be more similar to the correct option and thus are more likely than other options to be chosen. While dependencies between the choices can easily be incorporated in cognitive models, that would result in a large number of free parameters that would make fitting to data difficult. Conversely, RTNet inherently learns all relationships between the choice options during the training of the Bayesian neural network that forms its core. RTNet still requires the fitting of the overall signal strength (which we accomplish by adjusting the noise level of the images fed to RTNet), but this single free parameter allows it to capture all choice option dependencies, something that traditional models cannot achieve.

Biological plausibility of neural network models of response time

Physiological recordings have uncovered several features of the processing in the human visual system that are relevant to judging the plausibility of the networks examined here. First, the conduction from one area to another in the visual cortex (roughly corresponding to different layers in neural networks) takes approximately 10 ms[48], with signal from the photoreceptors reaching the top of the visual hierarchy in inferior temporal cortex in 70-100 ms[49]. Therefore, a single sweep from input to output in a purely feedforward network should result in decisions with RT less than a few hundred milliseconds even though human decisions can range from a hundred of milliseconds to a few seconds. Second, neurons in each layer of the visual cortex continue to fire action potentials for hundreds of milliseconds after the stimulus onset and receive strong recurrent input from later layers of processing[50]. Finally, neuronal processing is known to be noisy such that the same image input generates very different neuronal activations on different trials[29].

MSDNet diverges from these known properties of the human visual cortex in several important ways. To generate meaningful RTs, MSDNet assumes that classification decisions are made after each layer of processing, though there is no evidence that decisions in the brain can be directly based on information in early visual cortex without further processing in subsequent layers. Moreover, because it assumes the existence of a single feedforward sweep through the network, it cannot naturally capture large RT variability between stimuli given the short latencies of processing between different layers. Finally, MSDNet does not incorporate any recurrent processing, capture the noisiness of the responses in the visual cortex, or replicate the long periods of activity of the neurons in each processing area. These properties strongly limit the biological plausibility of MSDNet.

In comparison, the dynamics of CNet are closer to those of biological neural networks. Indeed, several of CNet's features – such as parallel and continuous processing of input, and transmission delays between layers – were directly inspired by biology. The transmission delays allow the network to mimic the processing latencies across cortical layers. These features were also found to account for differences in processing efficiency between images such that CNet produced more rapid responses for prototypical images with clear backgrounds compare to unusual or cluttered images. However, CNet includes several features that are not biologically plausible such as its lack of stochasticity of decisions and recurrent processing. Further, it remains unclear how its cascaded architecture could map onto brain areas[10].

It is possible to introduce stochasticity in CNet and MSDNet by feeding the outputs of the final softmax layer into a race model. However, such an architecture would imply that response stochasticity arises purely from noise in the decision stage. Although decision noise may exist in humans contributing to noisy motor responses, stochasticity in human responses is thought to predominantly arise from noisy inference[21] or noisy sensory representations[51–53]. Therefore, CNNs with additional noise at the decision stage are less biologically plausible than RTNet, which includes noise in the evidence processing stage.

On the other hand, while also not capturing all properties of visual processing, RTNet appears more biologically plausible. First, it mimics the noisiness of neuronal responses for repeated presentations of the same stimulus. Second, because RTNet processes each stimulus multiple times, it naturally generates long-lasting neuronal activations and RTs on the order of many hundreds of milliseconds (or even seconds). Third, the network's output is inherently stochastic, unlike feedforward networks or MSDNet and CNet that are inherently deterministic. Finally, the accumulation process implemented in RTNet has been observed in multiple regions in the human parietal cortex, frontal cortex, and subcortical areas[54–57]. Nevertheless, one critical limitation of the biological plausibility of RTNet is its lack of recurrency. That being said, the question of how to train recurrent neural networks on static images remains open[44,49,58–60]. Further, while the core of RTNet does not include recurrency, the evidence accumulation system can be thought of as a recurrent network. In fact, several recent studies demonstrated the advantages of combining a standard feedforward network with a recurrent network in performing a range of tasks and extrapolating to solve problems of greater complexity than they were trained on[61,62]. Thus, while RTNet remains less biologically plausible than a true recurrent network, it is as biologically plausible as current methods of training neural networks permit.

Using noisy weights to generate stochasticity in RTNet's responses

One critical feature of RTNet is that its weights are noisy. Practically, there are many different ways of generating noise in the weights. In early iterations of RTNet, we attempted to create variability by training a feedforward network and then adding the same amount of variability to each connection. This approach resulted in variability that was too small for some weights and too large for others[63], often leading to no accuracy gains from the process of evidence accumulation. Indeed, a given amount of noise over a specific weight may not change the performance of a network at all, but the same disturbance over another weight may have destructive effects[64–66]. We therefore chose to obtain the weight variability by training a Bayesian neural network so that each weight has an appropriate amount of noise. In the future, it may be possible to use other methods for setting the noise level for each connection, but we are currently unaware of any method besides training a Bayesian neural network that can generate appropriate noise for each weight.

RTNet assumes that every time evidence is sampled from a stimulus, the network's weights change randomly (according to the BNN's posterior weight distributions). These random moment-by-moment fluctuations in the network's weights lead to noisy activations. However, in the brain, noisy activations in response to a stimulus are thought to arise from random fluctuations in neuronal activity itself. Therefore, it can be argued that a more biologically plausible implementation of RTNet would involve noise in unit activations rather than weights[67]. The main reason we chose to add noise in weights rather than activations is due to the practical ease of implementing BNNs that can naturally generate variability in networks. Mechanistically, however, there may be no meaningful distinction between noisy weights and noisy activations, since noisy weights lead to noisy activations, which mimic the randomness of neural responses.

Limitations

One limitation of RTNet is that its mechanism for stopping the accumulation process is non-optimal. Following a large literature of race models in cognitive psychology[16,34,47], RTNet makes a decision when any one choice option receives sufficient evidence to exceed a threshold. However, if another choice option has almost same amount of evidence, the observer has little ability to differentiate between the two choices and is essentially guessing between them. Previous research showed that guessing can be an appropriate behavior if the observer knows that the task is very difficult[68] or if the observer has been deliberating for a long time[69]. However, in a race model, guessing can happen at any time point regardless of task difficulty. Nevertheless, human decisions are often suboptimal[70,71], and therefore it is unclear as to whether this suboptimal decision-making mechanism should be seen as a drawback if the goal is to model human decision-making.

Another limitation of RTNet is that each sweep of the feedforward path is independent of the previous states, whereas the current state in the human brain is influenced by its previous states[58]. To address this limitation, the sampling process in RTNet can be modified such that the current state of the network depends on the previous states. For example, a weight over an edge at a specific moment can be made a function of its previous values, which would make the sequential samples dependent on each other. Additional studies are needed to investigate the effect of such state dependence on model performance.

Conclusion

We developed a new neural network, RTNet, which exhibits the basic features of human perceptual decision making and predicts human accuracy, RT, and confidence on an image-by-image basis. The network provides a better model of human perceptual decisions than the current state-of-the-art networks for generating response times – CNet and MSDNet. RTNet thus represents an important step in the use of neural networks as models of human decisions.

Methods

Behavioral experiment

Pre-registration

This study's sample size, experiment design, included variables, hypothesis, and planned analyses were pre-registered on Open Science Framework (https://osf.io/kmraq) prior to any data being collected.

Subjects

Sixty-four subjects (31 female, age=18-32) with normal or corrected to normal vision were recruited. We had pre-registered the collection of only 40 subjects, but due to less time restrictions than we had anticipated, and to further increase the statistical power, we collected data from more subjects. All subjects signed informed consent and were compensated for their participation. The protocol was approved by the Georgia Institute of Technology Institutional Review Board. All methods were carried out in accordance with relevant guidelines and regulations.

Stimulus, task, and procedure

Subjects performed a digit discrimination task where they reported their perceived digit followed by rating their decision confidence. Each trial began with subjects fixating on a small white cross for 500-1000 ms, followed by a presentation of the stimulus for 300 ms (**Figure *2***). The stimulus was a digit between 1 and 8 (the digits 0 and 9 were excluded) superimposed on a noisy background. Subjects' task was to report the perceived digit using a computer keyboard by placing four fingers of their left hand on numbers 1-4 and placing four fingers of their right hand on numbers 5-8. This setup allowed subjects to respond without looking at the keyboard, thus providing less noisy response times. Following their categorization response, subjects reported their decision confidence on a 4-point scale (where 1 corresponds to the lowest confidence and 4 corresponds to the highest confidence). There was no deadline on the response or confidence rating.

The experiment included manipulations of speed-accuracy trade-off and task difficulty. Speed-accuracy trade-off was manipulated by asking subjects to emphasize either the speed or accuracy of their responses. To facilitate proper responding, we organized the experiment into alternating blocks of speed and accuracy focus. Task difficulty was manipulated by adding different levels of uniform noise to the stimuli. Specifically, “easy” stimuli included average uniform noise of 0.25 (range = 0-0.5), whereas “difficult” stimuli included average uniform noise of 0.4 (range = 0-0.8). To add the noise, the pixel values were first transformed to be between 0 and 1 and random numbers drawn from the corresponding noise distributions were added separately to each pixel. We scaled the resulting image to be between 0 and 1 again, and finally converted the image to a uint8 format (scaled between 0 and 255). The noise levels were chosen based on the pilot testing to produce two different performance levels. Easy and difficult images were randomly interleaved.

The task stimuli were selected from a publicly available handwritten digits (MNIST) dataset[24]. This dataset contains 60,000 training images and 10,000 testing images. Since the training images were used to train the models in this study, we randomly selected images from MNIST test set to include in our experiment. This ensures that the selected images for the experiment are novel both for the human subjects and for the trained models. We randomly selected 480 images for the experiment (120 for each condition). The MNIST dataset images are of size 28 x 28 pixels which appeared overly small on the computer screens we were using. Therefore, before adding noise, the selected images were first resized to 84 x 84 pixels (using MATLAB's *imresize* function), and they were padded with the background color of MNIST images to size 256 x 256 pixels (visual angle = 6.06°).

The experiment started with three blocks of training each containing 50 trials. The first block contained images from the MNIST dataset without any noise. This was done to familiarize the subjects with the experiment. The next two blocks were used to introduce the speed-accuracy trade-off by asking subjects to focus on accuracy in the first block and on speed in the second. The difficulty level of the stimuli in these two training blocks was same as in the main experiment. During the whole training session, the experimenter was standing beside the subject quietly and was available to answer any questions. None of the images used in the training session was used in the main experiment.

Once the subject confirmed that he or she understands the task, the experimenter left the room and subjects completed the main experiment that consisted of 960 trials organized in four runs each containing four blocks of 60 trials. Each block consisted of a single speed-accuracy trade-off condition, and each run included exactly two "accuracy focus" and two "speed focus" conditions in a randomized order. At the beginning of each block, subjects were given the name of the condition for that block ("accuracy focus" or "speed focus") and asked to adjust their responding policy accordingly. In each block, we pseudo-randomly interleaved trials from the two difficulty levels such that each was presented exactly 30 times. All 480 images were shown to subjects in first two runs and the procedure was repeated with a new random ordering of the stimuli in the last two runs. All images were same for all subjects, and each image was assigned only to one specific condition.

Apparatus

The experiment was designed in MATLAB 2020b environment using Psychtoolbox 3[72]. The stimuli were presented on a 21.5-inch Dell P2217H monitor (1920 x 1080 pixel resolution, 60 Hz refresh rate). Subjects were seated 60 cm away from the screen and provided their responses using a keyboard.

Behavioral analyses

We followed the data analyses steps outlined in our preregistration. We first excluded subjects who did not follow sufficiently well the speed/accuracy instructions by not providing faster average RT in the "speed focus" compared to the "accuracy focus" condition. This resulted in removing two subjects (out of 64). We preregistered the exclusion of subjects with floor or ceiling effects on accuracy but no subject met the criteria for exclusion. However, following our preregistration, we excluded two subjects because they showed ceiling effects for confidence. Note that our preregistration document called for excluding subjects who provided average confidence of more than 3.7 but because this would have resulted in excluding a much larger number of subjects than we had anticipated, we only excluded subjects whose average confidence was above 3.85. Therefore, 60 subjects were used in all subsequent analyses.

We additionally excluded individual trials with extreme RT values using preregistered criteria based on Tukey's interquartile criterion. Specifically, for each subject, we computed the 25th and 75th percentiles of the RT distributions in each condition. We then removed all RTs with values more than 1.5 times the interquartile range such that if $Q1$ is the RT value at the 25th percentile and $Q3$ is the RT value at the 75th percentile, we removed values smaller than $Q1 - 1.5 \times (Q3 - Q1)$ and larger than $Q3 + 1.5 \times (Q3 - Q1)$. This step resulted in removing an average of 5.46% of total trials (range of 1.35-8.22% for each subject).

Once these preprocessing steps were completed, we computed average accuracy, RT, confidence, and skewness of the RT distributions separately for each condition. The skewness was computed as $\frac{\sum_{i=1}^{N}(x_i-\mu)^3}{(N-1)\sigma^3}$ where μ and σ are the mean and standard deviation of the sample distribution, respectively. We also computed average RT and average confidence scores for error and correct trials across subjects to examine how RT and confidence change as a function of response accuracy. Finally, for visualization purposes, we plotted RT distributions for one subject in Figure 4C. The RT distributions were generated using kernel density estimation (KDE), which approximates the underlying probability density function that generated the data by smoothing the observations with a Gaussian kernel[73]. The KDE plots were created using Seaborn's KDE plot with a smoothing bandwidth of 1.2[74].

Network architecture

RTNet

The RTNet model consists of two main modules (**Figure *1*B**). The first module is a Bayesian neural network (BNN) which is capable of making predictions regarding an image. BNNs are a type of artificial neural network built by introducing stochastic components into the network to simulate multiple possible models with their associated probability distribution[75]. The main difference between a BNN and standard feedforward neural network is that in BNN the weights are distributions instead of point estimates. A random sample from these distributions results in a unique feedforward network. This random sampling enables variability in the output of the network, which in turn can be fed into an accumulation process that drives a decision. The second module of our model consists of exactly such accumulation of the evidence produced on each step by the first module. At each processing step, the output of the network (in the form of activations of the final layer) was accumulated towards a pre-defined threshold. Evidence for each choice option was accumulated separately from the rest, similar to a race model[16]. The accumulation process continues until the total amount of accumulated evidence for one of the alternatives reaches a predefined threshold. The alternative for which the threshold was reached then becomes the response of the model. The response time produced by RTNet is simply the number of samples used to reach the decision threshold. The confidence of the model was obtained by taking the difference in evidence scores between the chosen response and the second-best choice.

CNet

The parallel cascaded network (CNet) builds upon the architecture of residual networks (ResNet) by utilizing skip connections to introduce propagation delays during input processing. At each processing step, all units in all layers are updated parallelly. However, due to the propagation delays introduced by each residual block, simpler perceptual features get transmitted faster across blocks. For instance, at the first time-step, only the first residual block receives input and model predictions at this step are based only on the computations of the first residual block. At the second time step, all the other layers receive partial input from the first block. Even though the model prediction at this point will be based on computations from all blocks, only the first block will have received complete input and achieved stable output. The other blocks will only contain partial updates from the lower block and therefore their output will not be stable. In general, a residual block, t, takes $(t-1)$ time steps to receive complete and stable input. At any point during processing, the network can generate a prediction since all the residual blocks contribute to the computations. However, if the time step (t) is less than the number of residual blocks, the responses will be based on unstable representations in the higher blocks. Due to this architecture, the network's responses are subject to a trade-off between speed and complexity of processing. Decision time is indicated by the processing step at which the decision was made, and decision confidence is derived from the softmax value in the final layer, at the time of decision. The softmax values are obtained by transforming the activation scores (z) of all nodes in the output layer according to the function: $\frac{e^{z_i}}{\sum_j^n e^{z_j}}$, where i refers to the node whose output is being transformed and n refers to the number of nodes in the output layer (which is equal to the number of classes).

MSDNet

MSDNet has an architecture similar to a standard feedforward neural network (**Figure *1*A**) but with early-exit classifiers after each of its layers (**Figure *1*C**). At each output layer, the evidence for each choice is computed using a softmax function and if the evidence for any alternative exceeds a predefined value the network stops processing and immediately produces a response. The layer at which the response was made is indicative of the decision time, and the softmax value at that layer is indicative of decision confidence[76,77].

Implementation

We implemented both RTNet and MSDNet using the AlexNet architecture, which has eight layers with learnable parameters[25]. The AlexNet architecture consists of five convolutional layers with a combination of max pooling followed by three fully connected layers. For MSDNet, in addition to the standard AlexNet structure, we incorporated additional readout layers located right after each layer of processing (**Figure *1*C**). The feature map size of all these readout layers were set to the number of classes.

CNet was implemented using the architecture of ResNet-18[7] since it requires networks with skip connections. ResNet-18 architecture consists of 17 convolutional layers, where 16 of these layers are embedded within eight residual blocks (skip connections), followed by a final fully-connected layer with softmax activation to generate the decision.

We chose to implement RTNet within a relatively large-scale CNN such as AlexNet (rather than a shallow network which may have also been able to learn to classify the MNIST dataset). Our goal was to eventually compare our model to others such as CNet and MSDNet, which are generally based on larger CNNs and work on multiple existing datasets. Additionally, difficulties associated with training Bayesian neural networks limited us to relatively small network structures (rather than VGG or ResNet models). We found the AlexNet architecture to be a reasonable compromise in this trade-off between model complexity and ease of training BNNs. All neural networks were implemented in PyTorch[78]. Bayesian networks were implemented using Pyro[79], which is a probabilistic programming library built on PyTorch.

Network training

We trained all the models to achieve classification accuracy higher than 97% on the MNIST test set.

RTNet

We trained the BNN module of RTNet for a total of 15 epochs with a batch size of 500. We used the Evidence lower bound (ELBO) loss function[80] and Adam[81] for optimization with a learning rate of 0.001, and the default values for weight decay and epsilon (weight decay = 0; ϵ = 10^{-8}). To ensure that each network performs greater than 97% on MNIST test set, we followed a specific rule for each model. When testing an image with the BNN module of RTNet, we sampled 10 times from the posterior distributions learned during the training and thus obtained 10 unique responses for each image. The response with highest frequency among 10 responses was chosen as the final decision of the BNN module. We resized the MNIST images to the standard input size to Alexnet model architecture (227 x 227 pixels). We also normalized the input images to have a mean of 0.1307 and standard deviation of 0.3081, which is a standard procedure when using Alexnet for classification of the ImageNet dataset[82]. We trained sixty instances of RTNet using the above procedure but with different weight initializations for each network instance. We used a different combination of mean and standard deviation (SD) values for each of the 60 instances to maximize differences in network initializations. Specifically, different network instances of RTNet were initialized such that all means of the weights and biases were set to a value between 0.1 and 1.2 with 0.1 increments, and all SDs of weights and biases were set to a value ranging from 1 to 5 with increments of 1 (for a total of $12 \times 5 = 60$ instances).

CNet

We trained CNet using the same procedure that was used by the original authors since their training protocol was found yielded the best network behavior and performance. The network achieved an accuracy > 97% with 12 training epochs and a batchsize of 500. The models were trained on a temporal-difference (TD) learning procedure along with cross-entropy loss. In the original publication, TD learning was found to perform better than softmax-based cross-entropy loss in encouraging correct responses to emerge faster. The loss function was optimized using an initial learning rate of 0.01, weight decay of 0.005 and a momentum of 0.9. The images were normalized to a mean of 0.1307 and standard deviation of 0.3081. We trained sixty instances of

CNet using the above procedure but using a different random seed for initializing the network's weights to allow individual differences in network's learning.

MSDNet

Due to its deterministic nature, for MSDNet, only three epochs with a batchsize of 500, were enough to achieve test accuracy of more than 97% with the same batch size and a weighted cumulative loss function[76]. Adam[81] was used for optimization with a learning rate of 0.001. For testing, the response of the last output layer was taken as the network's decision. If a network did not achieve accuracy greater than 97%, we started the training over with the same initial values. Since MSDNet is also built on the architecture of AlexNet, we resized the MNIST images to the standard input size for AlexNet and normalized the images to have a mean of 0.1307 and standard deviation of 0.3081. To make the initializations of MSDNet as similar as possible to the initializations of RTNet, for each RTNet instance, we set the initial values for the weights and biases of the MSDNet instance by randomly sampling from the Gaussian distribution used in the corresponding RTNet initialization.

Choosing parameters that allow the models to mimic human accuracy

Because the goal of our study was to examine whether the models exhibit the signatures of human perceptual decision making, we matched the accuracy of the models across the four experimental conditions to the average accuracy in the human data. For all models, this was achieved by adjusting the noise level in the images (separately for the "easy" and "difficult" images) and the threshold parameter (separately for the speed and accuracy conditions). Lower noise levels lead to higher accuracy, whereas higher threshold parameters lead to longer processing and response times (and also contribute to higher accuracy levels).

Parameter values were adjusted using a coarse search followed by a fine search. In the coarse search for RTNet, we varied the amplitude of uniform noise from 1 to 10 with increments of 1 (where the noise amplitude refers to the length of the interval over which the noise values are generated), and the threshold value from 2 to 12 with increments of 2. The results were closest to the human accuracy levels when the noise was in the range 2-3 for easy images and 4-5 for difficult images, and the threshold was set to 2-4 for the speed focus condition and 6-8 for the accuracy focus condition. We then conducted a fine search near those values by changing the noise level from 2 to 5 with 0.1 increments and changing the threshold values from 2 to 8 with 0.5 increments. The closest match to human accuracy was achieved for noise levels of 2.1 and 4.1 for easy and difficult images, respectively, and a threshold value of 3 for the speed condition and 6 for the accuracy condition.

We used a similar procedure to tune the parameters of CNet. Note that the threshold value for CNet is the softmax evidence at the output layer. The coarse search was performed using threshold values between 0.5 and 0.9 with increments of 0.04. The results were closest to the human accuracy levels when the threshold was in range 0.79-0.83 for the speed focus condition, and 0.86-0.9 for the accuracy focus condition. We then performed a fine search in these ranges by incrementing the threshold by steps of 0.01. The closest match to human accuracy was achieved for a threshold value of 0.83 for the speed condition and 0.9 for the accuracy condition. For noise levels, the best match to human accuracy was obtained when the noise levels were set to 1.42 and 1.83 for easy and difficult images, respectively.

We also used a similar procedure to tune the parameters of MSDNet. Note that the threshold value for MSDNet is the softmax evidence at each early exit. The coarse search was performed using the threshold values between 0.5 and 0.95 with increments of 0.05. The results were closest to the human accuracy levels when the threshold was in range 0.55-0.65 for the speed focus condition, and 0.8-0.9 for the accuracy focus condition. We then performed a fine search in these ranges by incrementing the threshold by steps of 0.01. The closest match to human accuracy was achieved for a threshold value of 0.58 for the speed condition and 0.82 for the accuracy condition. For finding the optimal noise levels, the best match was obtained when the noise levels were set to 1.9 and 3.0 for easy and difficult images, respectively.

Data and code availability

Behavioral data, as well as all codes and trained models are publicly available at: https://osf.io/akwty.

References

1. Ratcliff, R. A theory of memory retrieval. *Psychol Rev* **85**, 59–108 (1978).
2. Ratcliff, R. & McKoon, G. The diffusion decision model: theory and data for two-choice decision tasks. *Neural Comput* **20**, 873–922 (2008).
3. Kriegeskorte, N. Deep Neural Networks: A New Framework for Modeling Biological Vision and Brain Information Processing. *http://dx.doi.org/10.1146/annurev-vision-082114-035447* **1**, 417–446 (2015).
4. Kriegeskorte, N. & Golan, T. Neural network models and deep learning. *Current Biology* **29**, R231–R236 (2019).
5. Kietzmann, T. C., McClure, P. & Kriegeskorte, N. Deep Neural Networks in Computational Neuroscience. *Oxford Research Encyclopedia of Neuroscience* (2019) doi:10.1093/ACREFORE/9780190264086.013.46.
6. Yamins, D. L. K. & DiCarlo, J. J. Using goal-driven deep learning models to understand sensory cortex. *Nature Neuroscience 2016 19:3* **19**, 356–365 (2016).
7. Iuzzolino, M. L., Mozer, M. C. & Bengio, S. Improving Anytime Prediction with Parallel Cascaded Networks and a Temporal-Difference Loss. *Adv Neural Inf Process Syst* **33**, 27631–27644 (2021).
8. Spoerer, C. J., Kietzmann, T. C., Mehrer, J., Charest, I. & Kriegeskorte, N. Recurrent neural networks can explain flexible trading of speed and accuracy in biological vision. *PLoS Comput Biol* **16**, (2020).
9. Zhang, L. *et al.* SCAN: A Scalable Neural Networks Framework Towards Compact and Efficient Models. *Adv Neural Inf Process Syst* **32**, (2019).
10. Subramanian, A., Sizikova, E., Kumbhar, O., Majaj, N. & Pelli, D. G. Benchmarking dynamic neural-network models of the human speed-accuracy tradeoff. *J Vis* **22**, 4359–4359 (2022).
11. Huang, G. *et al.* Multi-Scale Dense Networks for Resource Efficient Image Classification. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings* (2017).
12. Kalanthroff, E., Davelaar, E. J., Henik, A., Goldfarb, L. & Usher, M. Task conflict and proactive control: A computational theory of the Stroop task. *Psychol Rev* **125**, 59–82 (2018).
13. Mewhort, D. J. K., Braun, J. G. & Heathcote, A. Response Time Distributions and the Stroop Task: A Test of the Cohen, Dunbar, and McClelland (1990) Model. *J Exp Psychol Hum Percept Perform* **18**, 872–882 (1992).
14. Cohen, J. D., Dunbar, K. & McClelland, J. L. On the control of automatic processes: a parallel distributed processing account of the Stroop effect. *Psychol Rev* **97**, 332–361 (1990).
15. Bogacz, R., Brown, E., Moehlis, J., Holmes, P. & Cohen, J. D. The physics of optimal decision making: A formal analysis of models of performance in two-alternative forced-choice tasks. *Psychol Rev* **113**, 700–765 (2006).
16. Heathcote, A. & Matzke, D. Winner takes all! What are race models, and why and how should psychologists use them? *Curr Dir Psychol Sci* (2022).

17. Vickers, D. Evidence for an Accumulator Model of Psychophysical Discrimination. *http://dx.doi.org/10.1080/00140137008931117* **13**, 37–58 (2007).
18. Forstmann, B. U., Ratcliff, R. & Wagenmakers, E.-J. Sequential Sampling Models in Cognitive Neuroscience: Advantages, Applications, and Extensions. *Annu Rev Psychol* **67**, 641–66 (2016).
19. Rahnev, D. Confidence in the Real World. *Trends Cogn Sci* **24**, 590–591 (2020).
20. Yeon, J. & Rahnev, D. The suboptimality of perceptual decision making with multiple alternatives. *Nature Communications 2020 11:1* **11**, 1–12 (2020).
21. Drugowitsch, J., Wyart, V., Devauchelle, A. D. & Koechlin, E. Computational Precision of Mental Inference as Critical Source of Human Choice Suboptimality. *Neuron* **92**, 1398–1411 (2016).
22. Li, H. H. & Ma, W. J. Confidence reports in decision-making with multiple alternatives violate the Bayesian confidence hypothesis. *Nat Commun* **11**, 1–11 (2020).
23. Churchland, A. K., Kiani, R. & Shadlen, M. N. Decision-making with multiple alternatives. *Nat Neurosci* **11**, 693 (2008).
24. Deng, L. The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Process Mag* **29**, 141–142 (2012).
25. Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems 25 (NIPS 2012)* https://proceedings.neurips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html (2012).
26. Geirhos, R. *et al.* Generalisation in humans and deep neural networks. *Advances in Neural Information Processing Systems 31* https://proceedings.neurips.cc/paper/2018/hash/0937fb5864ed06ffb59ae5f9b5ed67a9-Abstract.html (2018).
27. Geirhos, R. *et al.* Comparing deep neural networks against humans: object recognition when the signal gets weaker. (2017) doi:10.48550/arxiv.1706.06969.
28. Beck, J. M., Ma, W. J., Pitkow, X., Latham, P. E. & Pouget, A. Not Noisy, Just Wrong: The Role of Suboptimal Inference in Behavioral Variability. *Neuron* **74**, 30–39 (2012).
29. Renart, A. & Machens, C. K. Variability in neural activity and behavior. *Curr Opin Neurobiol* **25**, 211–220 (2014).
30. Heitz, R. P. The speed-accuracy tradeoff: history, physiology, methodology, and behavior. *Front Neurosci* **8**, 150 (2014).
31. Heitz, R. P. & Schall, J. D. Neural mechanisms of speed-accuracy tradeoff. *Neuron* **76**, 616–28 (2012).
32. Ratcliff, R. & Rouder, J. N. Modeling Response Times for Two-Choice Decisions. *Psychol Sci* **9**, 347–356 (1998).
33. Wagenmakers, E.-J. & Brown, S. On the Linear Relation Between the Mean and the Standard Deviation of a Response Time Distribution. *Psychol Rev* **114**, 830–841 (2007).
34. Brown, S. & Heathcote, A. The simplest complete model of choice response time: Linear ballistic accumulation. *Cogn Psychol* **57**, 153–178 (2008).
35. Forstmann, B. U. *et al.* Striatum and pre-SMA facilitate decision-making under time pressure. *Proceedings of the National Academy of Sciences* **105**, 17538–17542 (2008).

36. Luce, R. D. *Response Times*. (Oxford University Press, 1986). doi:10.1093/acprof:oso/9780195070019.001.0001.
37. Ratcliff, R. A diffusion model account of response time and accuracy in a brightness discrimination task: fitting real data and failing to fit fake but plausible data. *Psychon Bull Rev* **9**, 278–291 (2002).
38. Rahnev, D. Visual metacognition: Measures, models, and neural correlates. *Am Psychol* **76**, 1445–1453 (2021).
39. Wyart, V. & Koechlin, E. Choice variability and suboptimality in uncertain environments. *Curr Opin Behav Sci* **11**, 109–115 (2016).
40. Findling, C. & Wyart, V. Computation noise in human learning and decision-making: origin, impact, function. *Curr Opin Behav Sci* **38**, 124–132 (2021).
41. Gold, J. I. & Shadlen, M. N. The Neural Basis of Decision Making. (2007) doi:10.1146/annurev.neuro.29.051605.113038.
42. Yeung, N. & Summerfield, C. Metacognition in human decision-making: confidence and error monitoring. *Philosophical Transactions of the Royal Society B: Biological Sciences* **367**, 1310–1321 (2012).
43. Heathcote, A. & Love, J. Linear deterministic accumulator models of simple choice. *Front Psychol* **3**, 292 (2012).
44. Spoerer, C. J., Kietzmann, T. C., Mehrer, J., Charest, I. & Kriegeskorte, N. Recurrent neural networks can explain flexible trading of speed and accuracy in biological vision. *PLoS Comput Biol* **16**, e1008215 (2020).
45. Ratcliff, R. & Smith, P. L. A Comparison of Sequential Sampling Models for Two-Choice Reaction Time. *Psychol Rev* **111**, 333–367 (2004).
46. Brown, S. & Heathcote, A. A ballistic model of choice response time. *Psychol Rev* **112**, 117–128 (2005).
47. Tillman, G., Van Zandt, T. & Logan, G. D. Sequential sampling models without random between-trial variability: the racing diffusion model of speeded decision making. *Psychon Bull Rev* **27**, 911–936 (2020).
48. Mizuseki, K., Sirota, A., Pastalkova, E. & Buzsáki, G. Theta Oscillations Provide Temporal Windows for Local Circuit Computation in the Entorhinal-Hippocampal Loop. *Neuron* **64**, 267–280 (2009).
49. Nayebi, A. *et al.* Task-Driven Convolutional Recurrent Models of the Visual System. *Adv Neural Inf Process Syst* **2018-December**, 5290–5301 (2018).
50. Issa, E. B., Cadieu, C. F. & Dicarlo, J. J. Neural dynamics at successive stages of the ventral visual stream are consistent with hierarchical error signals. *Elife* **7**, (2018).
51. Kaufman, M. T. & Churchland, A. K. Sensory noise drives bad decisions. *Nature 2013 496:7444* **496**, 172–173 (2013).
52. Brunton, B. W., Botvinick, M. M. & Brody, C. D. Rats and humans can optimally accumulate evidence for decision-making. *Science* **340**, 95–98 (2013).
53. Osborne, L. C., Lisberger, S. G. & Bialek, W. A sensory source for motor variation. *Nature 2005 437:7057* **437**, 412–416 (2005).
54. Huk, A. C. & Shadlen, M. N. Neural Activity in Macaque Parietal Cortex Reflects Temporal Integration of Visual Motion Signals during Perceptual Decision Making. *Journal of Neuroscience* **25**, 10420–10436 (2005).

55. Huk, A. C., Katz, L. N. & Yates, J. L. The Role of the Lateral Intraparietal Area in (the Study of) Decision Making. *Annu Rev Neurosci* **40**, 349 (2017).

56. Bahl, A. & Engert, F. Neural circuits for evidence accumulation and decision making in larval zebrafish. *Nature Neuroscience 2019 23:1* **23**, 94–102 (2019).

57. Hanks, T. D., Kiani, R. & Shadlen, M. N. A neural mechanism of speed-accuracy tradeoff in macaque area LIP. *Elife* **2014**, (2014).

58. van Bergen, R. S. & Kriegeskorte, N. Going in circles is the way forward: the role of recurrence in visual inference. *Current Opinion in Neurobiology* vol. 65 176–193 Preprint at https://doi.org/10.1016/j.conb.2020.11.009 (2020).

59. Spoerer, C. J., McClure, P. & Kriegeskorte, N. Recurrent Convolutional Neural Networks: A Better Model of Biological Object Recognition. *Front Psychol* **0**, 1551 (2017).

60. Kietzmann, T. C. *et al.* Recurrence is required to capture the representational dynamics of the human visual system. *Proceedings of the National Academy of Sciences* **116**, 21854–21863 (2019).

61. Schwarzschild, A. *et al.* Can You Learn an Algorithm? Generalizing from Easy to Hard Problems with Recurrent Networks. *Advances in Neural Information Processing Systems 34* https://proceedings.neurips.cc/paper/2021/hash/3501672ebc68a5524629080e3ef60aef-Abstract.html (2021).

62. Zhou, D. *et al.* Least-to-Most Prompting Enables Complex Reasoning in Large Language Models. (2022) doi:10.48550/arxiv.2205.10625.

63. Saltelli, A. *et al.* Sensitivity Analysis for Neural Networks. Natural Computing. *Risk Analysis* **159**, 179–201 (2009).

64. Ko, J. H., Kim, D., Na, T., Kung, J. & Mukhopadhyay, S. Adaptive weight compression for memory-efficient neural networks. *Proceedings of the 2017 Design, Automation and Test in Europe, DATE 2017* 199–204 (2017) doi:10.23919/DATE.2017.7926982.

65. Koutník, J., Gomez, F. & Schmidhuber, J. Evolving neural networks in compressed weight space. *Proceedings of the 12th Annual Genetic and Evolutionary Computation Conference, GECCO '10* 619–625 (2010) doi:10.1145/1830483.1830596.

66. Kung, J., Kim, D. & Mukhopadhyay, S. A power-aware digital feedforward neural network platform with backpropagation driven approximate synapses. *Proceedings of the International Symposium on Low Power Electronics and Design* **2015-September**, 85–90 (2015).

67. Tsvetkov, C., Malhotra, G., Evans, B. D. & Bowers, J. S. The role of capacity constraints in Convolutional Neural Networks for learning random versus natural data. *Neural Netw* **161**, 515–524 (2023).

68. Malhotra, G., Leslie, D. S., Ludwig, C. J. H. & Bogacz, R. Overcoming indecision by changing the decision boundary. *J Exp Psychol Gen* **146**, 776 (2017).

69. Drugowitsch, J., Moreno-Bote, R. N., Churchland, A. K., Shadlen, M. N. & Pouget, A. The Cost of Accumulating Evidence in Perceptual Decision Making. *Journal of Neuroscience* **32**, 3612–3628 (2012).

70. Rahnev, D. & Denison, R. N. Suboptimality in perceptual decision making. *Behavioral and Brain Sciences* **41**, (2018).

71. Evans, N. J., Bennett, A. J. & Brown, S. D. Optimal or not; depends on the task. *Psychon Bull Rev* **26**, 1027–1034 (2019).

72. Brainard, D. H. The Psychophysics Toolbox. *Spat Vis* **10**, 433–436 (1997).

73. Chen, Y. C. A tutorial on kernel density estimation and recent advances. *https://doi.org/10.1080/24709360.2017.1396742* **1**, 161–187 (2017).

74. Waskom, M. L. seaborn: statistical data visualization. *J Open Source Softw* **6**, 3021 (2021).

75. Jospin, L. V., Buntine, W., Boussaid, F., Laga, H. & Bennamoun, M. Hands-on Bayesian Neural Networks -- a Tutorial for Deep Learning Users. (2020).

76. Kumbhar, O., Sizikova, E., Majaj, N. & Pelli, D. G. Anytime Prediction as a Model of Human Reaction Time. (2020).

77. Huang, G. *et al.* Multi-Scale Dense Networks for Resource Efficient Image Classification. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings* (2017).

78. Paszke, A. *et al.* PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Advances in Neural Information Processing Systems 32 (NeurIPS)* https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html (2019).

79. Bingham, E. *et al.* Pyro: Deep Universal Probabilistic Programming. *Journal of Machine Learning Research* **20**, 1–6 (2019).

80. Kingma, D. P. & Welling, M. Auto-Encoding Variational Bayes. *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings* (2013) doi:10.48550/arxiv.1312.6114.

81. Kingma, D. P. & Ba, J. L. Adam: A Method for Stochastic Optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings* (2014) doi:10.48550/arxiv.1412.6980.

82. Deng, J. *et al.* ImageNet: A large-scale hierarchical image database. 248–255 (2010) doi:10.1109/CVPR.2009.5206848.