Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

LETTER

Neural Response Time Analysis: XAI Using Only a Stopwatch†

J. Eric T. Taylor*[1] | Shashank Shekhar[1,2] | Graham W Taylor[1,2,3]

[1]Vector Institute, Ontario, Canada
[2]School of Engineering, University of Guelph, Ontario, Canada
[3]Canada CIFAR AI Chair

Correspondence
*Email: eric.taylor@vectorinstitute.ai

Present Address
661 University Ave, Suite 710, Toronto, ON, Canada, M5G 1M1

Summary

How would you describe the features that a deep learning model composes if you were restricted to measuring observable behaviours? Explainable artificial intelligence (XAI) methods rely on privileged access to model architecture and parameters that is not always feasible for most users, practitioners, and regulators. Inspired by cognitive psychology research on humans, we present a case for measuring response times (RTs) of a forward pass using only the system clock as a technique for XAI. Our method applies to the growing class of models that use input-adaptive dynamic inference and we also extend our approach to standard models that are converted to dynamic inference post hoc. The experimental logic is simple: If the researcher can contrive a stimulus set where variability among input features is tightly controlled, differences in response time for those inputs can be attributed to the way the model composes those features. First, we show that RT is sensitive to difficult, complex features by comparing RTs from ObjectNet and ImageNet. Next, we make specific *a priori* predictions about RT for abstract features present in the SCEGRAM dataset, where object recognition in humans depends on complex intra-scene object-object relationships. Finally, we show that RT profiles bear specificity for class identity, and therefore the features that define classes. These results cast light on the model's feature space without opening the black box.

KEYWORDS:
Explainable Artificial Intelligence, Convolution Neural Networks, Dynamic Inference, Inductive Bias, Hypothesis Validation, Object Classification, Scene Grammar

1 | INTRODUCTION

The majority of techniques developed for XAI depend on privileged access to the architecture and parameters of the model in question[1]. If XAI as a field is to provide satisfying explanations for decisions and behaviours to end users, researchers will need ways to generate explanations from "outside" the black-box — without having to inspect the model internals, rather, by interacting with it by curating inputs and observing outputs. Explanations from outside the black box are desirable because they empower any user to investigate the cause and consequence of otherwise inscrutable model processes.

The black-box problem in XAI is similar to the challenge faced in building explainable models of an analogous black box — the human mind. Some machine learning researchers have already begun to model AI decisions using methods from cognitive psychology (e.g.[?] [2,3,4,5,6,7,8,9,10,11].) Cognitive psychology is characterized by inferring hidden mental processes from observed

†Eric and Shashank contributed equally to this publication.
[0]**Abbreviations:** XAI, explainable artificial intelligence; RT, response time; NRT, neural response time

behaviour. XAI without privileged model access must also infer hidden processes from output, so cognitive psychology's methods are attractive when explanations are desired from outside the black box.

In humans, an easily-measured behaviour that corresponds to the complexity of mental operations is response time (RT)[12,13]. For example, difficult visual search tasks take more time[14]. RT methods could in principle be adapted to machines with only a simple system clock and stimuli curated to test a given hypothesis. However, unlike human brains, most deep learning models for computer vision (e.g.[15],[16]) perform a fixed number of operations over a static interval of time between receiving an input and yielding an output. This results in an uninformative RT distribution.

A new class of "dynamic inference" models offers a potential use case for RT methods. As deep learning models continue to grow and are deployed in resource and time-critical environments, there is a growing push towards models which can adapt their processing dynamically based on input complexity. This feature is desirable in embedded systems and autonomous vehicles, among other things. Initially proposed to address the issue of resource-efficiency[17], the scope of dynamic inference models has quickly grown to enable several desirable properties, such as semantic layout enabled models[18], adversarial robustness[19], or mitigating overthinking[20]. Not only does this class of models have a performance and resource advantage, they can provide sufficient variance in RT to draw meaningful conclusions for XAI. By modelling when to exit successfully in a hierarchical framework, dynamic inference models have "baked-in" interpretability into their decision making. Our approach, which we call Neural Response Time Analysis, exploits this underappreciated property of this class of models.

Given a dynamic inference model, our technique works by measuring only the softmax output and system clock time elapsed for a given input known to contain certain features. Thus, NRTA is agnostic to model architecture, its implementation, and the dynamic inference strategy used, making it adaptable to any use-case where variable RTs are available. We then analyse RTs recorded in response to carefully controlled input stimuli using both quantitative comparisons in values and trends of the NRT as well as by performing hypothesis testing on the nature of response across these stimuli.

Our first experiment compares RT profiles for input from ImageNet versus ObjectNet[21], which was designed to contain non-stereotyped features. Results from this experiment provide a useful proof of concept for using RT to make inferences about the complexity of a model's composed feature space. In addition, we demonstrate how intermediate classifiers can be trained and appended to any standard model after training, broadening the applications of RT analyses. The second experiment makes specific predictions about how dynamic inference models will process stimuli from the SCEGRAM dataset — a set of images specifically designed to display different values of the high-level, abstract notion of scene grammar studied in humans (e.g.[22]). We perform *a priori* hypothesis testing for when RT is and is not sensitive to scene grammar. The RT effects in Experiments 1 & 2 are observed despite insensitivity in accuracy measures, demonstrating that RT profiles can be more informative than item difficulty. Finally, we show that inputs with shared features exhibit RT profiles that follow similar trajectories. As such, RT profiles allow inferences about the model's feature space and can describe more than the difficulty of a prediction task.

2 | BACKGROUND

This work bridges several sub-fields within psychology and artificial intelligence. We briefly outline them in this section.

2.1 | Artificial Cognition for XAI

Most XAI techniques depend on privileged access to the model being explained. Some of these popular techniques involve: training an adjacent model to explain another using captioning[23], visualizing the gradient of the loss function with respect to the input at the penultimate layer[24], optimizing random input to visualize the "preferred" stimulus of a given neuron or layer[24,25], or systematically perturbing input to see which elements affect the output[26,27]. Although these methods are indispensable and ingenious, they share two vulnerabilities: (1) they can be difficult to implement from "outside" the black box, and (2) because they explore the space of explanations iteratively and automatically, interpretation depends on many observer degrees of freedom, whether through hyperparameter selection or through confirmation bias of the interpreter.

A complementary approach to XAI, adapted from experimentation in human cognitive psychology, shows how explanations can also be generated by testing specific *a priori* hypotheses of how models ought to behave under certain assumptions. Observing model behaviour under fair attempts to falsify the hypothesis results in intuitive explanations for how the model behaves. This is exactly how cognitive psychologists approach explaining the human mind, and it can be re-purposed for explaining any

black box. We form a falsifiable hypothesis for how the model ought to behave under a set of contrived and highly controlled experimental stimuli that vary only along the experimental dimension.

Studies inspired by the cognitive approach to understanding the mind are becoming more common in XAI. Ritter et al. devised an experiment to determine whether image classification models would rely on shape rather than colour — like humans — for one-shot object recognition[?]. Using two sets of matching stimuli: one matching in shape of the familiar object vs. other matching the color (and controlling all other low-level features), they found the model to more reliably identify the shape-matching stimulus as belonging to the same category. This was interpreted as demonstrating shape-bias in object learning. A similar line of hypothesis-testing driven research contrived different variations of same input either with/without features highlighted by a saliency algorithm for their PilotNet model, while controlling other variables[2]. PilotNet responded to untampered input in a similar manner as the salient input alone, and it did not respond to input with salient information removed. The behavioural results clearly supported the explanation that the information discovered by their saliency algorithm did in fact control steering. Geirhos et al. used a similar hypothesis-testing approach when they created a stimulus set that directly pitted texture and shape cues against each other to observe whether CNNs overindex on either dimension preferentially[4]. By creating mash-up stimuli that contained the global shape of one input and the texture of another, the CNNs were shown to reliably identify the stimuli according to their texture class.

Whereas most XAI techniques explore the range of possible explanations automatically (e.g. through optimizing for preferred input or systematic perturbation), we pit alternative explanations against each other from "outside" the black box. by measuring response times. Core to the approach is the process of contriving a test that may discern between competing hypotheses. We use stimulus sets that are known to contain different features, specifically non-stereotyped viewpoints (Experiment 1) and syntactically or semantically inconsistent object-object relationships (Experiment 2). Consequently, we can attribute differences in RTs between conditions of our experiment to these features.

2.2 | RT methods for explaining human behaviour

RT is an easily-measured external behaviour for humans that corresponds to the complexity of mental operations[12]. In the original demonstration from 1868, Donders showed that repeating a prepared syllable was about 75 m/s faster than repeating an unknown syllable. The mental process of classifying an incoming syllable takes a human neural network an additional serial processing step requiring 75 m/s. Although Donders' Subtractive Method has seen some major revisions in the intervening 150 years (e.g.[13,28]), the core logic persists. Assuming there is a relationship between processing time and hierarchical feature space or task complexity, we can apply a similar approach to the inference of deep learning black box processes. However, unlike human brains, most DNNs for computer vision perform a fixed number of operations over a static time interval, resulting in a uniform and uninformative RT distribution. If we want to use RT methods to explain DNN behaviours, we require a *distribution* of RTs.

2.3 | Input-Adaptive Dynamic Inference

Input-adaptive dynamic inference, also called conditional computation, or simply adaptive/dynamic inference, is an emerging paradigm for learning resource-efficient deep learning models with representative works proposed in[29,17,30,31,29,32,20,33]. These models allocate more computational resources and processing time to harder examples and less to "easy" examples during inference time. Thus the models do not need to sacrifice accuracy for efficiency at training time. Dynamic inference has become an increasingly popular approach to resource efficient ML due to its flexible speed-accuracy trade-off.

Among dynamic inference models, early-exit models[34,33,32,19] operate by adding auxiliary classifiers to intermediate layers. This allows for "early" and fast exits from neural network computations for simple instances, and later exits for harder instances. Huang et al.[34] utilized multi-scale features for early-exit models to reach near state-of-the-art performance on ImageNet[35] and other standard image recognition tasks. While most prior works have utilized softmax-based confidence scores for choosing their early exit criteria, more recent works explored other criteria like patience or change in intermediate predictions[36], and surprise or negative log-likelihood from a secondary auto-regressive model[37]. The early exits can be used as probes for interpretability to identify and mitigate "overthinking", the phenomena where deep neural nets reach correct decisions in intermediate classifiers but a wrong decision on further processing[20].

Most of the prior work on dynamic inference is focused on either designing efficient architectures[34] or better learning algorithms[38] that exploit input-adaptivity. Our approach is an orthogonal effort to explore the input-adaptive inductive bias as a

tool for interpretability. For our purposes, dynamic inference produces the key requirement for using RT methods: a distribution of response times corresponding to model depth and abstraction of feature space. We limit the scope of our NRT analysis to confidence-based early-exit models[34,20]. However, NRT analysis can directly be extended to other early-exit criteria which mitigate the issues with softmax-based confidence estimates[39] as well as other forms of dynamic inference models. Finally, while the scope of this study is black-box XAI for dynamic inference models, we also present a use-case for a universal extension of our technique. Based on[20] we also add early-exit probes to a standard ResNet-56 architecture to extend our analyses to *off-the-shelf* neural networks.

3 | MEASURING NEURAL RESPONSE TIMES

We studied the Multi-Scale DenseNet (MSDNet)[34] and Shallow Deep Network (SDN)[20], specifically the ResNet-56 variant) as our reference early-exit models. In terms of number of layers, the auxiliary internal classifiers (ICs) were equally spaced across the depth of the model following the same procedure as described in[34] and[20]. In addition to a final classifier, MSDNet and SDN have a total of 4 and 6 of these ICs respectively.

All models were trained on ImageNet using a sum of cross-entropy losses across all ICs and final classifier unless otherwise mentioned. The top-5 test accuracies for each model is reported across all the classifiers. MSDNet trained in this manner achieved accuracy of 80, 86.2, 88.6, 89.4, and 90.4%. For the SDN, we trained a standard ResNet-56 for 100 epochs, froze its parameters, added the ICs and trained only the IC parameters for another 25 epochs. This training strategy (IC-only) resulted in top-5 accuracies of 17.4, 20.6, 31.3, 34.9, 54.8, 67.8, and 80.2%. SDN was also trained end-to-end in a standard manner which led to an accuracy of 31.3, 35.2, 51.6, 59.6, 69.8, 76.9, and 78% respectively across the classifiers.

The models were implemented with the PyTorch[40] framework. To ensure that the RT statistics were model and implementation independent, we did not query the internals of the model architecture for calculating timing values. Instead, we relied entirely on the system clock time elapsed during the forward pass of a test input. The system used to perform the RT calculations had a 2 × E5-2620 v2 Hex-core processor (12 , physical CPU cores) and 128 GB RAM. Inference was performed on each input stimuli using two 11 GB NVIDIA 2080Ti graphics processing units. We avoided spurious correlations in RTs across stimuli by passing each input individually during test time, as is commonly done for evaluation of dynamic inference methods.

The models decide to make an “early exit" based on some confidence threshold or an anytime prediction[34] at an arbitrary time. The classifiers’ top-5 softmax estimates were used as a proxy for confidence; if the estimate was greater than the confidence threshold, it was recorded as sufficiently confident for an early exit. Thus, the classifier’s RT was recorded to be equal to the clock time at the moment of that classifier’s output minus the initial time. One of the core assumptions in our approach to RT methods is that this time will correspond directly to the number of operations. As each layer and classifier within our model performs a fixed number of operations, the RT to reach a decision for each classifier should be constant, with a small amount of error allowed for external variations in system processing. To verify this, we measured the correlation between FLOPs and RT to be $r = 0.996$. Instead of relying on exact measures of model operations, unavailable without access to the black-box, the RT measure is a very strongly correlated measure of the operations involved while only relying on the system clock time. Since the exact number of ICs is discernible from the number of discrete RT plateaus in the output, our method maintains a black-box approach for any dynamic inference model.

RT was modeled as a function of the range of confidence values (0 to 1 with a step size of .01) to fully profile how quickly the model would make decisions. For each image, the lowest auxiliary classifier with a decision at the given confidence was identified and its RT was saved. In some cases, higher classifiers made decisions with lower confidence than the preceding classifier, an artefact of the “overthinking" problem in deep neural networks[20]. In these cases, RT was recorded from the lower level as the early-exit architecture would prevent the higher classifier from ever making a decision. If the confidence threshold was higher than the confidence estimate across all exits, the RT was assigned to be the RT of the final classifier, since input processing will proceed up to the last exit. The full procedure is outlined in Algorithm 1.

4 | OBJECTNET - EXPERIMENT & RESULTS

To quantify the over-representation of canonical viewpoints and backgrounds in ImageNet, Barbu & Mayo et al. created ObjectNet, a 50,000-image test set for object recognition tasks that features common objects (many overlapping with ImageNet classes) viewed from non-canonical viewpoints on non-stereotyped backgrounds[21]. To achieve a more diverse set of features,

Algorithm 1: Time Series Step Function Generator

Require: $N \times K$ measurement matrix R reporting RT for each exit,
top-1 confidence $\theta = [\theta_1, \ldots, \theta_k, \ldots, \theta_K]$
for each classifier k,
confidence step size η

Result : $N \times S$ matrix R' representing RT to reach each confidence level

for all images, n
  for all steps, i
    for all auxiliary classifiers, k
      if $\theta_k > (i * \eta)$
        $R'_{n,i} = R_{n,k}$, **break**
      else if $\max_k \theta < (i * \eta)$
        $R'_{n,i} = R_{n,K}$

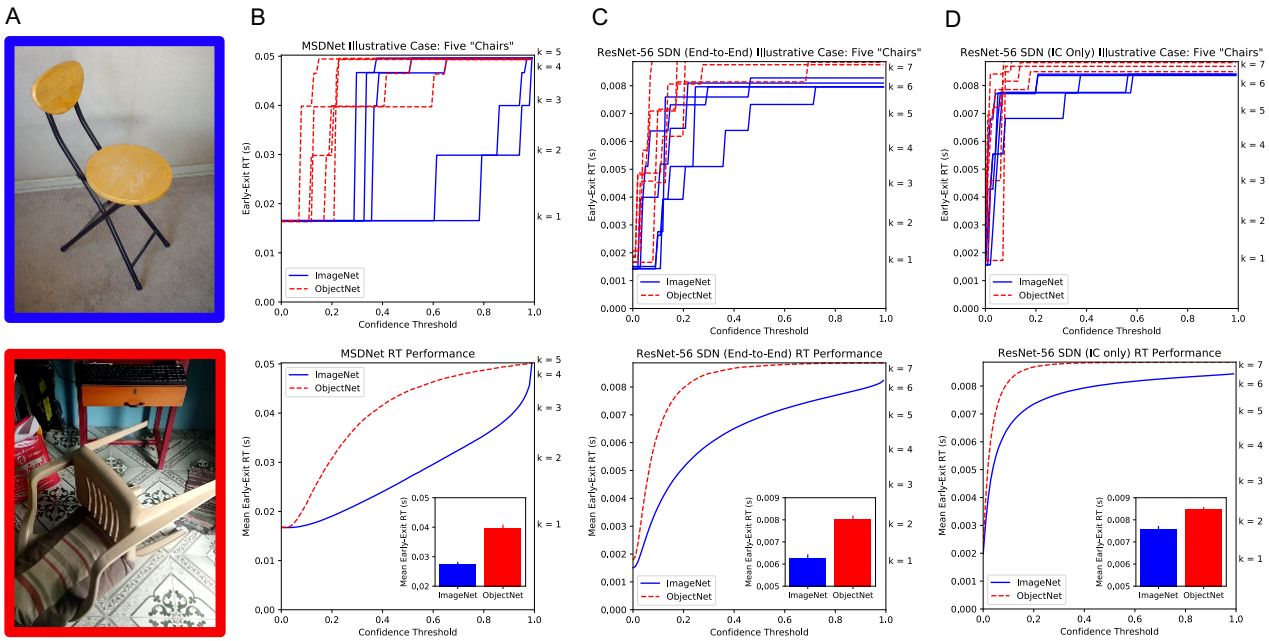


FIGURE 1 (A) Examples of chairs from ImageNet (blue) and ObjectNet (red). (B, above) Illustrative example of the step function describing the early-exit RT for five randomly-selected chairs from both ImageNet and ObjectNet. (B, below) Mean early-exit RT across all test images that had top-5 accuracy in the final auxiliary classifier. The values on the right vertical axis indicate the mean processing time for each of $K = 5$ auxiliary classifiers. Subplot displays the grand means for both datasets. (C) The same analyses for an alternate architecture, ResNet-56 SDN, trained end-to-end with $K = 7$ early exits. (D) The same analyses for ResNet-56 SDN, with $K = 7$ ICs appended after initial training. This shows that an off-the-shelf model can be converted and analyzed with NRT Analysis post hoc. Note differences in y-axis scale.

those authors recruited thousands of workers to photograph objects from specified angles communicated via smartphone. The result is a test set that is more inclusive of objects' non-canonical features. When popular object recognition models are tested on ObjectNet, they exhibit performance decrements of up to 45%.

Because ObjectNet is difficult for object recognition models, it is a candidate proof-of-concept test case that RT can be used to measure performance in DNNs. More importantly, because ObjectNet deliberately includes many non-canonical and presumably complex object features, which may be over-represented in later layers, it ought to display strong RT effects for models with hierarchical representation and dynamic inference, including MSDNet.

4.1 | MSDNet

RT was recorded from the 45,182 top-5-correct ImageNet images (90.3% top-5 accuracy) and 4,753 ObjectNet images that shared ImageNet labels and were also top-5 correct (25.6% top-5 accuracy on images with shared labels; while this is low, it is in the range of reported top-5 accuracy with overlapping labels by the ObjectNet team[21]). We analyzed inputs that were top-5 correct to ensure that we fairly compared object classes across datasets. Figure 1 illustrates how quickly MSDNet can make a decision given a range of confidence values between 0 and 1. Results from five randomly-selected images of chairs from both test sets are plotted to show how confidence propagates through the model, occasionally increasing RT in steps. The best performance would be a reverse-L shape, where the classifier $k = 1$ is sufficiently confident to identify the chair across the full range of thresholds.

RT was aggregated across all images in each set for every level of confidence. These values were submitted to an independent-samples t-test to affirm that RT can indeed be used as a reliable indicator of performance ($t = 9.29$, $p < .001$). Looking at the mean RT across all confidence thresholds, MSDNet processes ImageNet test stimuli 31.11% faster than ObjectNet stimuli with overlapping labels (27.40 ms vs. 39.77 ms). Because ObjectNet is characterized by a range of rotational viewpoints and non-canonical backgrounds, we infer that the higher-order features required to identify these items are better represented across MSDNet's auxiliary classifiers. These inputs are all top-5 correct, so it is worth noting that RT is sensitive to the discrepancy in feature space even when accuracy is not.

4.2 | ResNet56 SDN

Repeating the analyses as above, we found that both the end-to-end SDN ($t = 7.35$, $p < .001$) and the version with auxiliary classifiers appended after initial training ($t = 5.12$, $p < .001$) showed the same effect (although smaller in magnitude for the latter). The steeper curves indicate that the features required for correct classification are composed in deeper layers. Comparing across models, we can see that the end-to-end SDN composes the features required to achieve ∼50% confidence on ObjectNet a full IC earlier compared to the IC-only variant. This was expected since the IC-only model is initially a standard DNN optimized for the final classifier only later converted to a dynamic inference model. However, we chose to demonstrate the replication of our conclusion on it since this dramatically expands the usefulness of RT as an XAI technique beyond models with a baked-in input-adaptive inductive bias to any DNN. Going forward, we restrict our analyses to MSDNet.

5 | SCEGRAM - EXPERIMENT & RESULTS

The visual world is populated with regularities, enabling the learning of implicit rules for the relationships between objects and scenes[41]. Humans depend on scene grammar, or object-scene congruities, to guide perception and attention[42]. A semantic violation occurs when an object's identity is statistically uncorrelated with that of other scene elements (for instance, a roll of toilet paper placed in a microwave), whereas a syntactic violation occurs when the statistically reliable interposition of objects in a scene is upset (for instance, a roll of toilet paper placed in the bathroom sink). Evidence for the human phenomenon of scene grammar comes from studies showing that these violations produce performance decrements on our ability to identify the gist of scenes, guide visual attention[43,22], and to plan movements and object interactions[44]. Scene grammar effects also occur in artificial neural networks, with evidence for decreased performance in object and scene classification[45]. If MSDNet composes higher-order object features representing these relationships, they ought to occur in deeper layers, manifesting as a slower RT.

The SCEGRAM database[46] is a set of images of 62 indoor scenes with carefully curated manipulations of scene grammar. For each scene, there are four images (see Figure 2): consistent scene grammar (CON), inconsistent semantics (SEM), inconsistent syntax (SYN), and inconsistent semantics and syntax (SEMSYN). The semantic and syntactic manipulations are fully crossed. So for a given scene, say a kitchen counter, there are four versions of the image: a pot in a pile of dishes (CON); a clock in a pile of dishes (SEM); a pot affixed to the dishwasher door (SYN); and a clock affixed to the dishwasher door (SEMSYN). All other visual features in the scene are identical, allowing for experimental inferences about the scene grammar manipulations. We profiled MSDNet object recognition performance on 248 SCEGRAM images, taking the same measurements as in the ObjectNet experiment. SCEGRAM object labels do not match ImageNet perfectly so we manually checked that classification was correct.

The experimental considerations embedded in SCEGRAM allow us to make powerful, specific predictions about how MSDNet should process these inputs with respect to its RT. If MSDNet composes higher-order object features in later layers, then

early-exit decisions should be made using coarse features. If high-confidence classification depends on processing higher-order object features such as inter-object/object-scene relationships, then images with inconsistent scene grammar ought to be processed slower on average than images with consistent scene grammar. SCEGRAM also carefully controls salience, modeled by state-of-the-art human attention models[47] and validated with human foveal fixation data[46], ensuring that the four test images within a scene do not differ in salience. Consequently, differences in model performance cannot be attributed to the minimal low-level differences between scenes. SCEGRAM also contains normative data for consistency ratings, to assure that the scene grammar conditions violate expected inter-object relationships from a human perspective. Because of these controls, we can confidently attribute differences in the model's RT to the independent variable: scene grammar — or the features that present scene grammar to MSDNet. Finally, because MSDNet was pre-trained for object recognition, we can predict that RT effects should be specific to semantic, rather than syntactic inconsistencies.

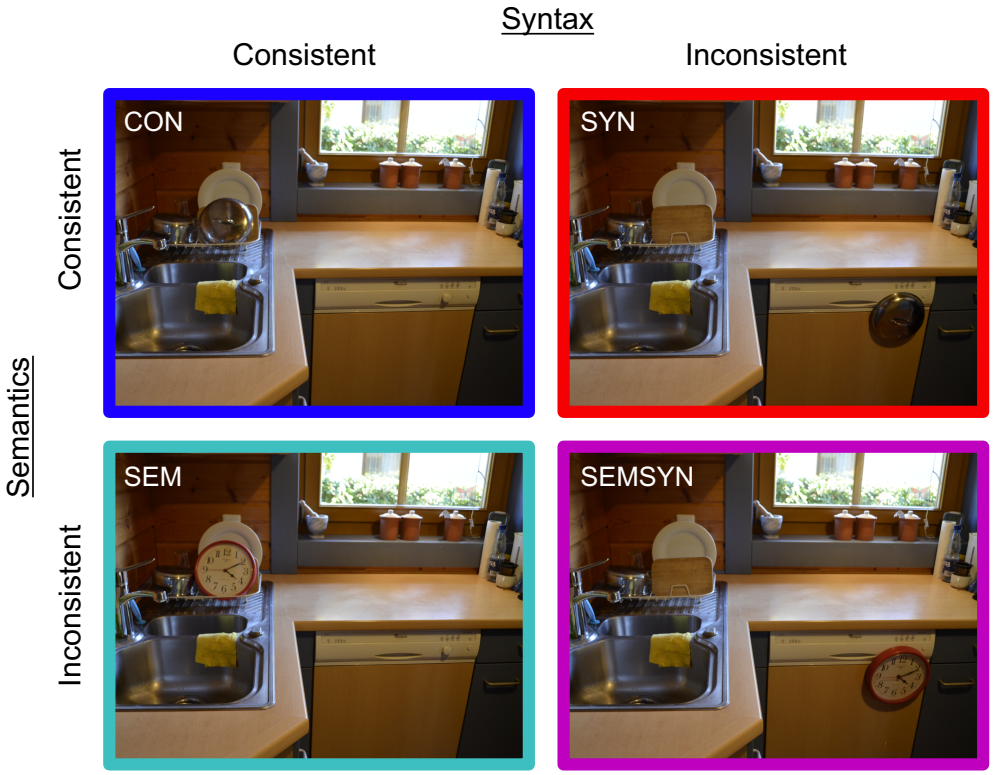


FIGURE 2 Illustrative example of SCEGRAM's test stimuli. Images feature two orthogonal manipulations: semantic consistency and syntactic consistency. A single scene is therefore presented to the model four times, with different combinations of scene grammar.

Step functions of RT given the full range of confidence thresholds are displayed in Figure 3. The mean early-exit RT across all 62 scenes for the full range of confidence thresholds was determined to generalize a profile of the relationship between RT, confidence, and scene grammar (see Figure 3B). Visual inspection of these means reveals better performance for scenes with consistent grammar with local troughs around 0.2 and 0.4 confidence. In human subjects experiments with electro-physiological or other time series data, it is common to specify a window of interest within which to compare RTs from different conditions[48]. We defined the boundaries for these windows as the mean confidence required by the model to reach the subsequent classifier's RT (e.g. if RT for $k = 2$ is 0.03 s, what is the mean confidence at which MSDNet reaches 0.03 s?). To characterize whether RT differences were reliable across scenes, we submitted the data to a three-way repeated-measures ANOVA with semantics (consistent, inconsistent), syntax (consistent, inconsistent), and classifier window (thresholds described above) as within-subjects factors. As expected, the classifier produced a strong effect on RT ($F(3,183) = 222.59$, $p < .001$), indicating that RTs were

slower as deeper layers were employed for visual recognition. The critical result is that, as expected, there was a significant effect of semantics ($F(1,61) = 4.87$, $p = .031$), indicating that SCEGRAM images with inconsistent semantic information were classified reliably slower than images with consistent semantics. No other effects reached statistical significance.

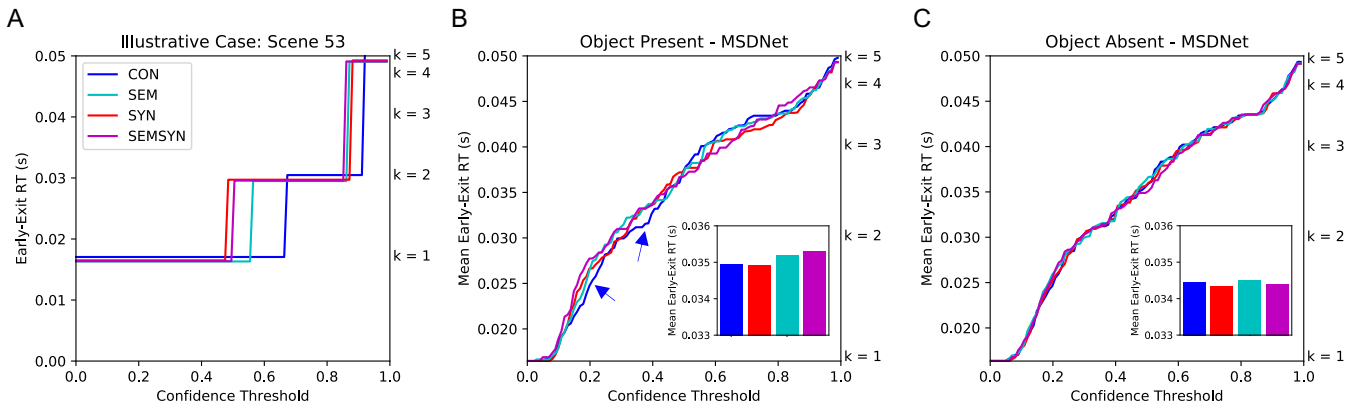


FIGURE 3 (A) Illustrative example of the step function describing the early-exit RT for a single scene in SCEGRAM. (B) Mean early-exit RT across all SCEGRAM scenes, grouped by scene grammar condition. Consistent scene grammar (CON) has visibly faster RT for responses around 0.2 and 0.4 confidence (blue arrows). The values on the right vertical axis indicate the mean processing time for each of $K = 5$ auxiliary classifiers. (C) Same analysis for the object-absent clone images in SCEGRAM. These are technically all semantically consistent. As predicted, they share the same RT profile as CON. Subplots display grand means for each condition. Best viewed in colour.

We can also predict when scene grammar effects should not emerge on RT. RTs were collected for the object-absent clone images corresponding to each of the same 248 SCEGRAM images used above. These images are identical except that the critical object has been removed, resulting in images with no inconsistencies. As predicted, there was no effect of semantic inconsistency ($F(1,61) = 0.36$, $p = 0.55$).

Analyzing the object-absent images provides some additional insight into the usefulness of RT measures. The object-present and object-absent images perform similarly in terms of recognition accuracy, despite the images having some strange objects in the inconsistent conditions. While top-5 accuracy failed to capture these oddities, the RT analysis revealed them. NRT analysis shows how model evaluation can explore other creative ways of documenting performance to capture subtleties in model processing.

6 | MASSIVE MEMORY - EXPERIMENT & RESULTS

A shrewd reader might rightfully point out that RT is conflated with classification difficulty in both previous experiments. It was essential to demonstrate that RT methods could capture differences in feature space, but these differences also correspond to difficulty; ObjectNet possesses non-stereotypical features, and that's what makes it harder and that's also what makes responses slower. Ideally, RT profiles can tell us about more than just difficulty. One approach to confirm this would be to show that RT profiles for inputs that rely on the same features for classification follow similar trajectories. To demonstrate this, we used the Massive Memory dataset[49], which contains images of 200 categories with 15 exemplars each. We asked whether RT profiles for inputs from the same class were more similar to each other than they were to RT profiles of inputs from other classes. This dataset is experimentally useful because the backgrounds have been omitted and background features that might otherwise complicate interpretation of the RT plots have been controlled for.

We profiled MSDNet performance on 3,000 Massive Memory dataset images (200 classes × 15 unique exemplars per class). RT data from each of the Massive Memory images was collected with the same process described in Section 4. We measured system clock time as RT, top 5 classes for each of $k = 5$ classifiers, and the confidence judgments of those classifications. All of these behaviours are observable from outside the black box. If inputs from the same class tend to share activations

throughout feature space, then RT profiles for same-class exemplars should follow similar trajectories compared to inputs from other classes. Step functions of RT given the full range of confidence thresholds are displayed in Figure 4A and 4B, illustrating all RT profiles for the inputs associated with two exemplar classes.

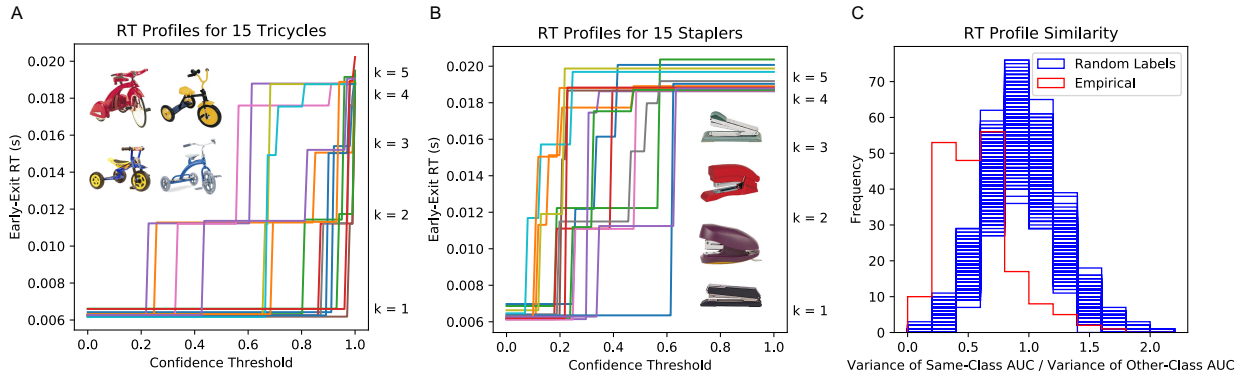


FIGURE 4 (A) Early-exit RTs as a function of confidence budget for all 15 tricycles in the Massive Memory dataset. Four examples displayed inset. (B) Early-exit RTs as a function of confidence budget for all 15 staplers in the Massive Memory dataset. Four examples displayed inset. The values on the left vertical axis indicate the mean processing time for each of $K = 5$ auxiliary classifiers. Visual inspection shows common trajectories for same-class exemplars. (C) This observation is quantified with a Monte Carlo analysis of a measure for assessing the same- versus other-class similarity of RT profiles. We calculated similarity as the variance of AUC for all inputs of a given class divided by the same variance for all inputs of all other classes. Empirical values for all 200 classes are depicted on a histogram in red. Lower values indicate more intra-class similarity of RT profiles. We repeated this analysis on 1,000 simulations of the same data with randomly permuted labels. Our empirical sample occurred well outside the distribution of all random samples.

To determine whether inputs with similar feature space follow similar RT profiles, we asked whether measures indicative of the RT profile's shape were more similar for same- versus different-class inputs. We calculated AUC for each individual RT profile and the ratio of the variance for AUC from inputs belonging to the same class versus variance for AUC for inputs from all other classes. This process was repeated for all 200 classes, and the distribution of these ratios are plotted in red in Figure 4C. Lower values on the x-axis indicate more similarity for AUC of same-class RT profiles compared with other-class.

To assess whether the observed distribution of our similarity metric could have occurred due to chance, we conducted a Monte Carlo simulation by randomly permuting labels in the Massive Memory dataset and repeating the above calculations with 1,000 such simulations. If there is no systematic trajectory for RT profiles of same-class inputs, randomly permuting the labels should have no effect on the observed distribution of our similarity metric.

Instead, we observed that all 1,000 simulated distributions resulting from randomly permuting labels occurred to the right of our empirical distribution. This is very strong evidence that the empirical distribution of our similarity metric did not occur due to chance. From this analysis, we conclude that RT profiles are indicative of the feature space, and therefore the class, of their input, and can theoretically be used to infer more than just the difficulty of classifying a particular input.

7 | CONCLUSIONS AND FUTURE WORK

We introduce NRT analysis as an XAI method and experimentally demonstrate its value by probing the inner workings of otherwise opaque models. We were able to test *a priori*, falsifiable hypotheses about the relationship between input space and response time using three different test sets. We showed that classification that depends on access to higher-layer features takes longer for dynamic inference models using conditional computation. These analyses could be used to form expectations for when and how models should perform in situations where explanations are desirable, but privileged access to a model is denied. The rate limiting factor on NRT analyses is the existence of test sets with meticulously controlled input features. Going forward, we would like to see the creation of more suitable test sets to expand the range of testable hypotheses.

ACKNOWLEDGMENTS

The authors would like to acknowledge Magdalena Sobol for her help with editing this article and Professor Ayesha Ali for guidance on the analysis in Experiment 3.

Author contributions

The manuscript was written by J. Eric T. Taylor and Shashank Shekhar with guidance from Graham W. Taylor. The models were trained and deployed and NRTs were collected by Shashank Shekhar. J. Eric T. Taylor analysed NRT data and visualized results. All authors contributed to study design and design of analyses.

Financial disclosure

None reported.

Conflict of interest

The authors declare no potential conflict of interests.

How to cite this article: J. E. T. Taylor, S. Shekhar, and G. W. Taylor (2021), Neural Response Time Analysis: XAI Using Only a Stopwatch, *Applied AI Letters*, .

APPENDIX

A ADDITIONAL OBJECTNET VS. IMAGENET RESULTS AND QUALITATIVE ANALYSIS

In Experiment 1, we showed that RT profiles could be used to account for differences in input difficulty that come with more complex feature space. We demonstrated this by showing individual RT profiles for five exemplars from one class in each dataset (chairs) and the mean RT across all input. In this supplementary Figure A1, we show a wider range of RT profiles from different classes so readers can witness some variability at the class level and gain confidence that the chair example was not cherry-picked. For the most part, inputs from ImageNet appear to be classified faster than the more difficult ObjectNet. In these plots, response time is plotted as a function of the confidence threshold. The function displays the minimum time required to make a classification at the given confidence. So a RT profile that plateaus at $k = 1$ until a confidence threshold of 0.5 indicates that the first intermediate classifier tops out at a softmax confidence of 0.5 and must proceed to the next intermediate classifier, taking more time. Inputs that can be confidently classified at earlier classifiers will have faster RTs.

In Figures A2 & A3, we display an illustrative case that shows how complex features may propagate through intermediate classifiers in time. These examples were selected to illustrate this relationship. ObjectNet is difficult for object classification because it contains images of objects seen from non-stereotypical viewpoints. When we examine the RT profiles, we access a new level of nuance. We can see how the "easy" examples cascade through the model over time, whereas the "hard" examples — viewed from different angles — are quickly abandoned by earlier classifiers. The authors of ObjectNet have stated that they intend to release rotational, viewpoint and scene annotations. When these are released, we can conduct a quantitative analysis examining these attributes' influence on RT.

B FULL RT PROFILES ON SCEGRAM AND ADDITIONAL NULL RESULTS

For completeness, Figure B4 displays RT profiles for every input in our analysis. The mean RT profiles for the four scene grammar conditions appear close to each other. This is expected, given that the dataset controls for salience and background features, and in some cases contains the same features but in different locations. Here, we wanted to display all individual RT profiles so readers could get a sense for the variability within the dataset.

In Experiment 2, we discussed how we could also predict when RT effects should *not* emerge when comparing inputs that varied in scene grammar. We used the object-absent control images from SCEGRAM as an example, but there are other cases where RT effects should not occur. We ran an identical analysis as reported in Experiment 2 using the SCEGRAM dataset but on a variant of MSDNet trained with an additional self-distillation loss based on[33]. The intermediate classifiers are trained to imitate the predictions of the final classifier, resulting in earlier classifiers that employ features like those composed in the final layers. This change in training would erase the correspondence between feature space and depth that RT methods depend on. Predictably, this analysis produced no significant differences in RT between scene grammar conditions (see Figure B5).

C INDIVIDUAL RT PROFILES FOR ALL MASSIVE MEMORY EXEMPLARS

In Experiment 3, we showed that inputs that share features, such as exemplars from the same class, will exhibit RTs that proceed through the range of confidence thresholds in a very similar manner. We supported this claim with a Monte Carlo simulation of label-permuted data, taking the variance of same-class AUC divided by the variance of all other-class AUC. For completeness, we show the individual RT profiles for the entire MM dataset so that readers can form intuition for how exemplars from the same class share RT profiles (see Figures C6-C10).

References

1. Rahwan I, Cebrian M, Obradovich N, et al. Machine behaviour. *Nature* 2019; 568(7753): 477–486. doi: 10.1038/s41586-019-1138-y

2. Bojarski M, Yeres P, Choromanska A, et al. Explaining how a deep neural network trained with end-to-end learning steers a car. *arXiv preprint arXiv:1704.07911* 2017.

3. Leibo JZ, d'Autume CdM, Zoran D, et al. Psychlab: A Psychology Laboratory for Deep Reinforcement Learning Agents. *arXiv:1801.08116 [cs, q-bio]* 2018. arXiv: 1801.08116.

4. Geirhos R, Rubisch P, Michaelis C, Bethge M, Wichmann FA, Brendel W. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231* 2018.

5. Kim B, Reif E, Wattenberg M, Bengio S. Do Neural Networks Show Gestalt Phenomena? An Exploration of the Law of Closure. *arXiv:1903.01069 [cs, stat]* 2019. arXiv: 1903.01069.

6. Roig G, Volokitin A, Poggio T. Do Deep Neural Networks Suffer from Crowding?. *Journal of Vision* 2018; 18(10): 902–902.

7. Osband I, Doron Y, Hessel M, et al. Behaviour suite for reinforcement learning. *arXiv preprint arXiv:1908.03568* 2019.

8. Gulordava K, Bojanowski P, Grave E, Linzen T, Baroni M. Colorless green recurrent networks dream hierarchically. *arXiv preprint arXiv:1803.11138* 2018.

9. Rajalingham R, Issa EB, Bashivan P, Kar K, Schmidt K, DiCarlo JJ. Large-scale, high-resolution comparison of the core visual object recognition behavior of humans, monkeys, and state-of-the-art deep artificial neural networks. *Journal of Neuroscience* 2018; 38(33): 7255–7269.

10. Richard Webster B, Yon Kwon S, Clarizio C, Anthony SE, Scheirer WJ. Visual psychophysics for making face recognition algorithms more explainable. In: ECVA. ; 2018: 252–270.

11. Henderson MM, Serences J. Biased orientation representations can be explained by experience with non-uniform training set statistics. *bioRxiv* 2020.

12. Donders FC. On the speed of mental processes. *Acta psychologica* 1868; 30: 412–431.

13. Sternberg S. The discovery of processing stages: Extensions of Donders' method. *Acta psychologica* 1969; 30(0): 276–315.

14. Treisman AM, Gelade G. A feature-integration theory of attention. *Cognitive psychology* 1980; 12(1): 97–136.

15. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* 2014.

16. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: IEEE. ; 2016: 770–778.

17. Bengio Y, Léonard N, Courville A. Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation. *arXiv:1308.3432 [cs]* 2013. arXiv: 1308.3432.

18. Andreas J, Rohrbach M, Darrell T, Klein D. Neural module networks. In: IEEE. ; 2016: 39–48.

19. Hu TK, Chen T, Wang H, Wang Z. Triple Wins: Boosting Accuracy, Robustness and Efficiency Together by Enabling Input-Adaptive Inference. In: ICLR. ; 2019.

20. Kaya Y, Hong S, Dumitras T. Shallow-deep networks: Understanding and mitigating network overthinking. In: PMLR. ; 2019: 3301–3310.

21. Barbu A, Mayo D, Alverio J, et al. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems* 2019; 32: 9453–9463.

22. Vo MLH, Henderson JM. Does gravity matter? Effects of semantic and syntactic inconsistencies on the allocation of attention during scene perception. *Journal of Vision* 2009; 9(3): 24–24. doi: 10.1167/9.3.24

23. Anne Hendricks L, Hu R, Darrell T, Akata Z. Grounding visual explanations. In: ECVA. ; 2018: 264–279.

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

25. Olah C, Mordvintsev A, Schubert L. Feature visualization. *Distill* 2017; 2(11): e7.

26. Zeiler MD, Fergus R. Visualizing and understanding convolutional networks. In: Springer. ; 2014: 818–833.

27. Ribeiro MT, Singh S, Guestrin C. " Why should i trust you?" Explaining the predictions of any classifier. In: ACM. ; 2016: 1135–1144.

28. Sternberg S, Knoll RL, others . The perception of temporal order: Fundamental issues and a general model. *Attention and performance IV* 1973: 629–685.

29. Wu Z, Nagarajan T, Kumar A, et al. Blockdrop: Dynamic inference paths in residual networks. In: ICCV. ; 2018: 8817–8826.

30. Graves A. Adaptive Computation Time for Recurrent Neural Networks. *arXiv:1603.08983 [cs]* 2017. arXiv: 1603.08983.

31. Wang X, Yu F, Dou ZY, Darrell T, Gonzalez JE. Skipnet: Learning dynamic routing in convolutional networks. In: ECVA. ; 2018: 409–424.

32. Zhang L, Tan Z, Song J, Chen J, Bao C, Ma K. SCAN: A Scalable Neural Networks Framework Towards Compact and Efficient Models. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R., eds. *Advances in Neural Information Processing Systems 32*Curran Associates, Inc. 2019 (pp. 4027–4036).

33. Phuong M, Lampert CH. Distillation-Based Training for Multi-Exit Architectures. In: IEEE. ; 2019: 1355–1364.

34. Huang G, Chen D, Li T, Wu F, Maaten v. dL, Weinberger KQ. Multi-Scale Dense Networks for Resource Efficient Image Classification. *arXiv:1703.09844 [cs]* 2018. arXiv: 1703.09844.

35. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. Imagenet: A large-scale hierarchical image database. In: Ieee. ; 2009: 248–255.

36. Zhou W, Xu C, Ge T, McAuley J, Xu K, Wei F. BERT Loses Patience: Fast and Robust Inference with Early Exit. *arXiv preprint arXiv:2006.04152* 2020.

37. Lugosch L, Nowrouzezahrai D, Meyer BH. Surprisal-Triggered Conditional Computation with Neural Networks. *arXiv preprint arXiv:2006.01659* 2020.

38. Li H, Zhang H, Qi X, Yang R, Huang G. Improved techniques for training adaptive deep networks. In: IEEE. ; 2019: 1891–1900.

39. Guo C, Pleiss G, Sun Y, Weinberger KQ. On calibration of modern neural networks. In: PMLR. ; 2017: 1321–1330.

40. Paszke A, Gross S, Massa F, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In: Wallach H, Larochelle H, Beygelzimer A, d'Alché-Buc F, Fox E, Garnett R., eds. *Advances in Neural Information Processing Systems 32*Curran Associates, Inc. 2019 (pp. 8024–8035).

41. Biederman I, Mezzanotte RJ, Rabinowitz JC. Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive Psychology* 1982; 14(2): 143–177. doi: 10.1016/0010-0285(82)90007-X

42. Võ MLH, Boettcher SE, Draschkow D. Reading scenes: how scene grammar guides attention and aids perception in real-world environments. *Current Opinion in Psychology* 2019; 29: 205–210. doi: 10.1016/j.copsyc.2019.03.009

43. Pereira EJ, Castelhano MS. Peripheral guidance in scenes: The interaction of scene context and object content.. *Journal of Experimental Psychology: Human Perception and Performance* 2014; 40(5): 2056. doi: 10.1037/a0037524

44. Draschkow D, Võ MLH. Scene grammar shapes the way we interact with objects, strengthens memories, and speeds search. *Scientific Reports* 2017; 7(1): 16471. doi: 10.1038/s41598-017-16739-x

45. Bayat A, Nand AK, Koh DH, Pereira M, Pomplun M. Scene Grammar in Human and Machine Recognition of Objects and Scenes. In: IEEE. IEEE; 2018; Salt Lake City, UT, USA: 2073–20737

46. Öhlschläger S, Võ MLH. SCEGRAM: An image database for semantic and syntactic inconsistencies in scenes. *Behavior research methods* 2017; 49(5): 1780–1791.

47. Walther D, Koch C. Modeling attention to salient proto-objects. *Neural networks* 2006; 19(9): 1395–1407.

48. Võ MLH, Wolfe JM. Differential Electrophysiological Signatures of Semantic and Syntactic Scene Processing. *Psychological Science* 2013; 24(9): 1816–1823. doi: 10.1177/0956797613476955

49. Konkle T, Brady TF, Alvarez GA, Oliva A. Conceptual distinctiveness supports detailed visual long-term memory for real-world objects.. *Journal of Experimental Psychology: General* 2010; 139(3): 558.

50. Helmholtz vH. *Messungen über den zeitlichen Verlauf der Zuckung animalischer Muskeln und die Fortpflanzungsgeschwindigkeit der Reizung in den Nerven: der physikalischen Gesellschaft zu Berlin mitgetheilt am 19. Juli 1850* . 1850.

51. Lillicrap TP, Kording KP. What does it mean to understand a neural network?. *arXiv:1907.06374 [cs, q-bio, stat]* 2019. arXiv: 1907.06374.

52. Gunning D. Explainable Artificial Intelligence (XAI) - DARPA. *machine learning* 2019: 18.

53. Gilpin LH, Bau D, Yuan BZ, Bajwa A, Specter M, Kagal L. Explaining Explanations: An Overview of Interpretability of Machine Learning. *arXiv:1806.00069 [cs, stat]* 2018. arXiv: 1806.00069.

54. Gunning D, Aha DW. DARPA's Explainable Artificial Intelligence Program. : 16.

55. Mueller ST, Hoffman RR, Clancey W, Emrey A, Klein G. Explanation in human-AI systems: A literature meta-review, synopsis of key ideas and publications, and bibliography for explainable AI. *arXiv preprint arXiv:1902.01876* 2019.

56. Lipton ZC. The Mythos of Model Interpretability. *arXiv:1606.03490 [cs, stat]* 2017. arXiv: 1606.03490.

57. Guidotti R, Monreale A, Ruggieri S, Turini F, Giannotti F, Pedreschi D. A Survey of Methods for Explaining Black Box Models. *ACM Computing Surveys* 2018; 51(5): 1–42. doi: 10.1145/3236009

58. Ritter S, Barrett DG, Santoro A, Botvinick MM. Cognitive psychology for deep neural networks: A shape bias case study. In: PMLR. ; 2017: 2940–2949.

59. Fancher R, Rutherford A. *Pioneers of psychology . New York: W. W.* Company Norton . 1979.

60. Grubb A, Bagnell J. SpeedBoost: Anytime Prediction with Uniform Near-Optimality. In: PMLR. ; 2012.

61. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. In: . 1. IEEE. ; 2001: I–I.

62. Viola P, Jones MJ. Robust real-time face detection. *International journal of computer vision* 2004; 57(2): 137–154.

63. Zhang C, Ren M, Urtasun R. Graph hypernetworks for neural architecture search. *International Conference on Learning Representations* 2019.

64. Yuan Z, Wu B, Liang Z, Zhao S, Bi W, Sun G. S2DNAS: Transforming Static CNN Model for Dynamic Inference via Neural Architecture Search. *arXiv preprint arXiv:1911.07033* 2019.

65. Hinton G, Vinyals O, Dean J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* 2015.

66. Huang G, Liu Z, Van Der Maaten L, Weinberger KQ. Densely connected convolutional networks. In: IEEE. ; 2017: 4700–4708.

67. Olah C, Satyanarayan A, Johnson I, et al. The Building Blocks of Interpretability. *Distill* 2018. https://distill.pub/2018/building-blocksdoi: 10.23915/distill.00010

68. Gunning D. Explainable artificial intelligence (xai). *Defense Advanced Research Projects Agency (DARPA), nd Web* 2017; 2(2).

69. Zhang Q, Yang Y, Liu Y, Wu YN, Zhu SC. Unsupervised learning of neural networks to explain neural networks. *arXiv preprint arXiv:1805.07468* 2018.

70. Si Z, Zhu SC. Learning and-or templates for object recognition and detection. *IEEE transactions on pattern analysis and machine intelligence* 2013; 35(9): 2189–2205.

71. Fitts PM. The information capacity of the human motor system in controlling the amplitude of movement.. *Journal of experimental psychology* 1954; 47(6): 381.

72. Taylor JET, Taylor GW. Artificial cognition: How experimental psychology can help generate explainable artificial intelligence. *Psychonomic Bulletin & Review* 2020: 1–22.

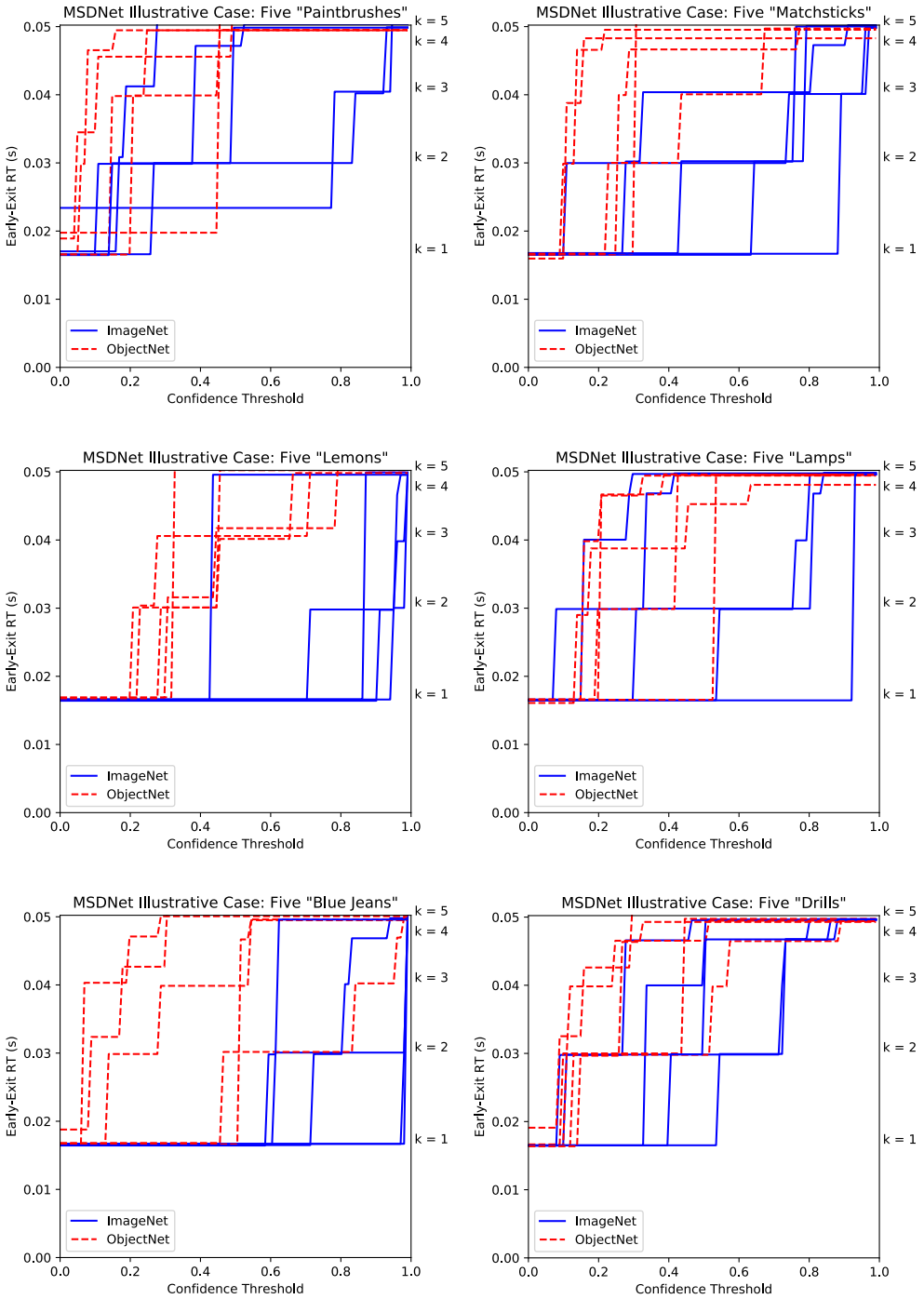


FIGURE A1 Additional illustrative examples of five exemplars for six different classes common to ImageNet and ObjectNet. These RT profiles show how quickly a classification can be made given a certain confidence threshold. The values on the right vertical axis indicate the mean processing time for each of $K = 5$ auxiliary classifiers. Throughout, we see that input from ImageNet is closer to the ideal reverse-L pattern that defines perfect performance (high confidence in the earliest classifier), whereas inputs from ObjectNet proceed to the intermediate classifiers faster given lower confidence thresholds.

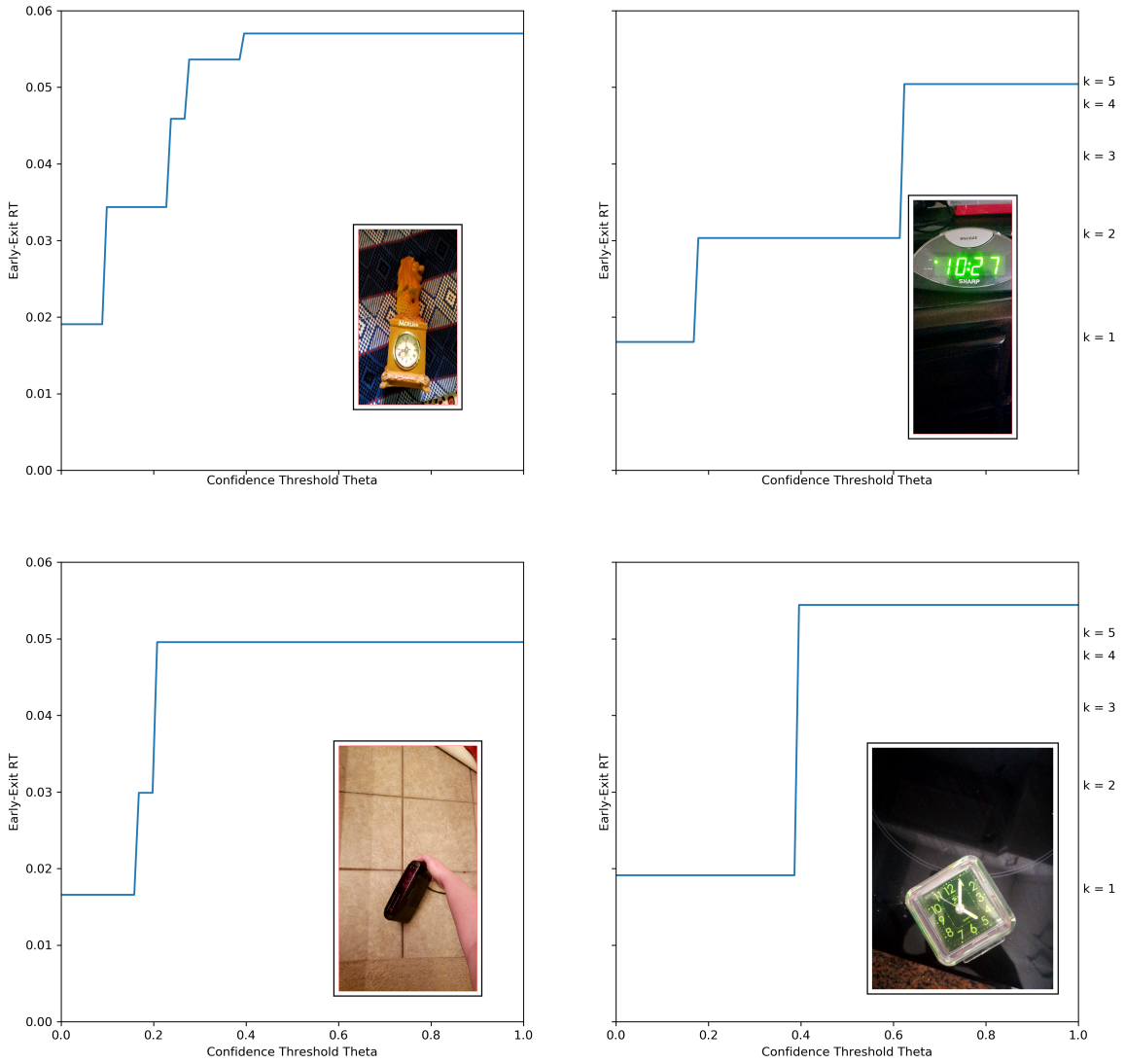


FIGURE A2 Exploration of relationship between RT and a complex feature — in this case, rotation. The top row shows two stereotypical viewpoints and the corresponding RT profiles. The bottom row shows RT profiles for non-stereotyped viewpoints of the same class.

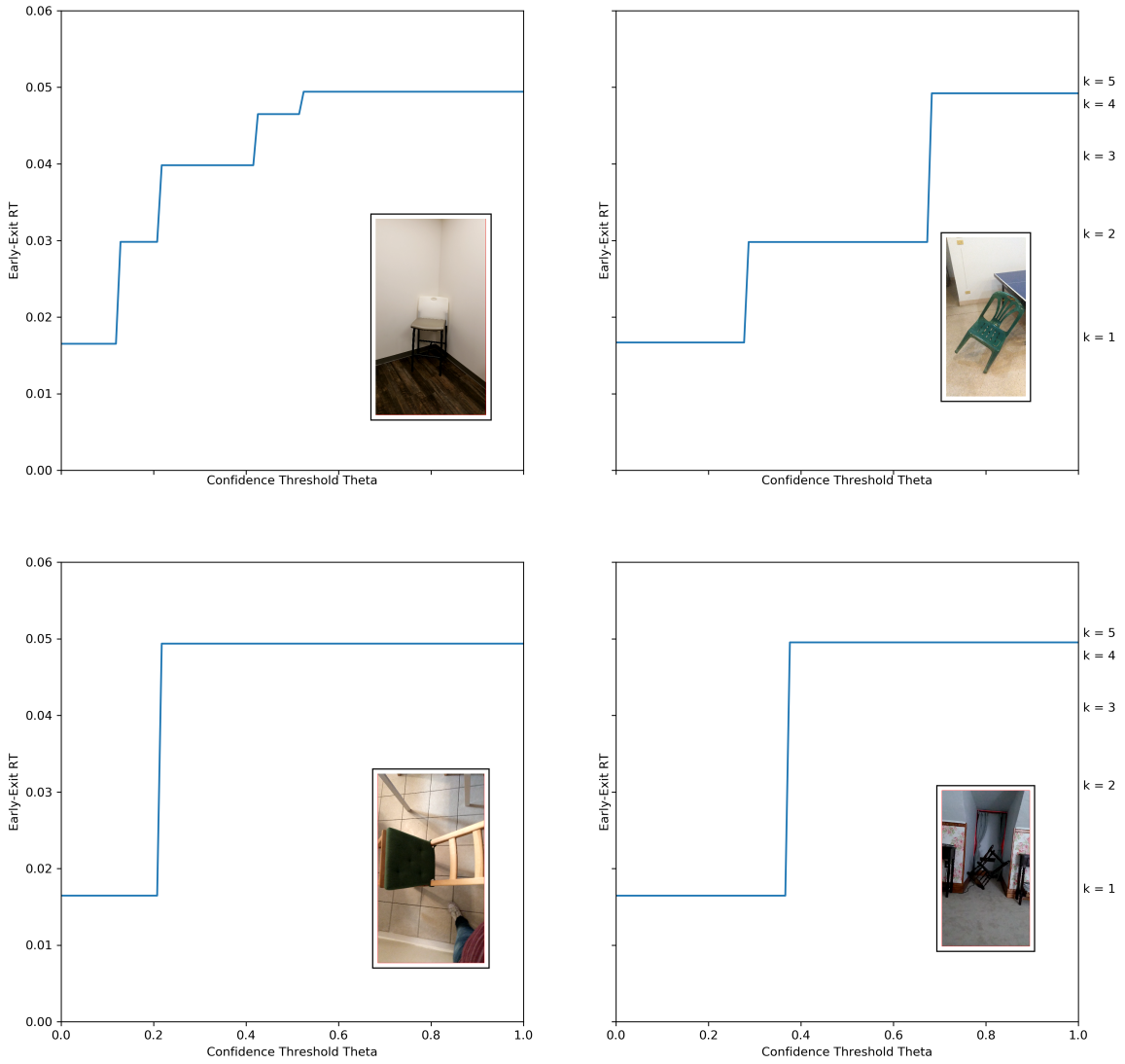


FIGURE A3 Exploration of relationship between RT and a complex feature — in this case, rotation. The top row shows two stereotypical viewpoints and the corresponding RT profiles. The bottom row shows RT profiles for non-stereotyped viewpoints of the same class.

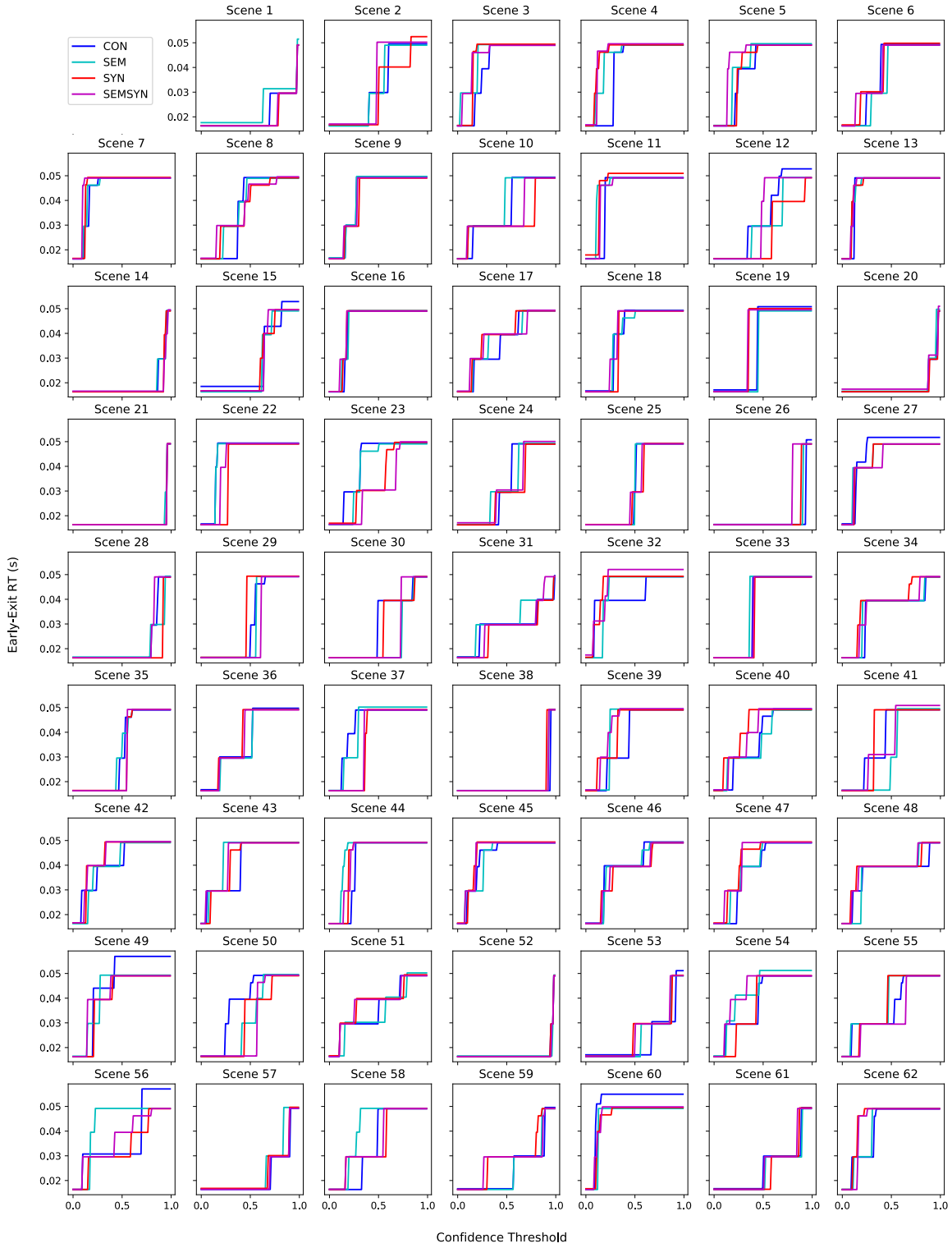


FIGURE B4 RT Profiles for all inputs from the SCEGRAM dataset used in Experiment 2, including all scene grammar conditions from all 62 scenes. These RT profiles show how quickly a classification can be made given a certain confidence threshold. The values on the right vertical axis indicate the mean processing time for each of $K = 5$ auxiliary classifiers.

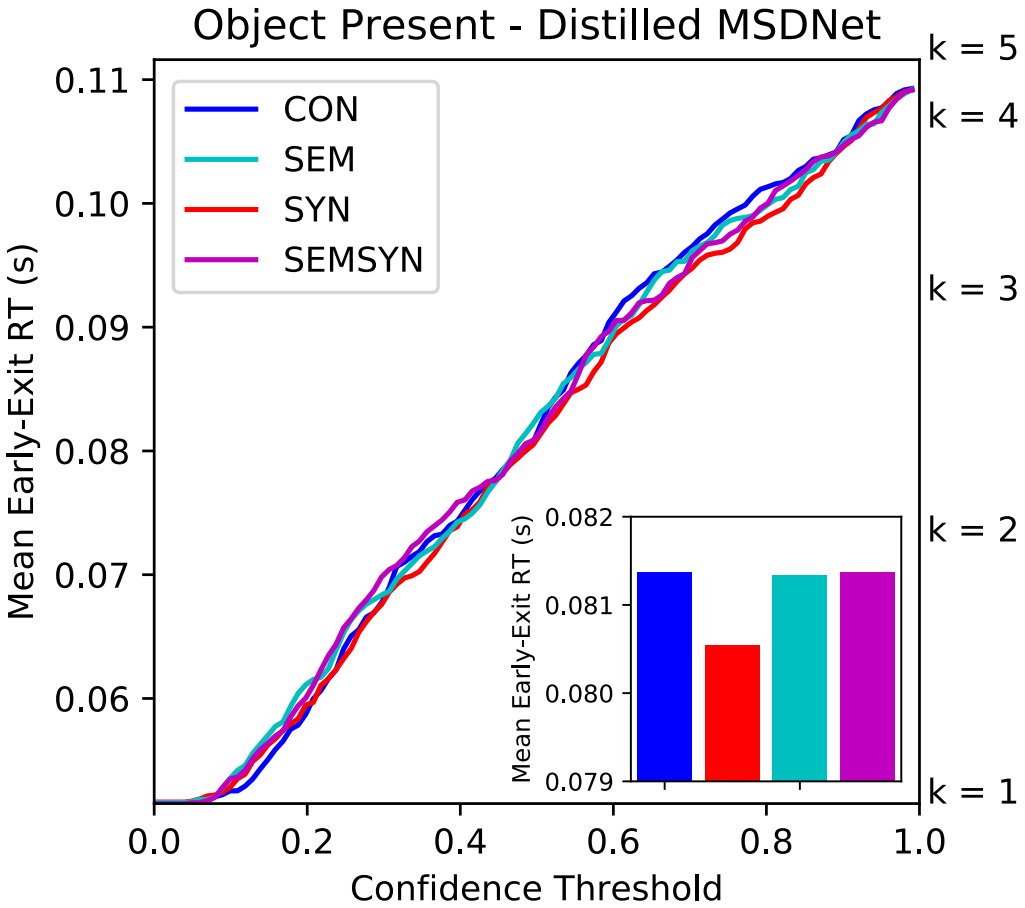


FIGURE B5 Mean early-exit RT across all SCEGRAM scenes, grouped by scene grammar condition for a distilled version of MSDNet. There are no statistically reliable differences between scene grammar conditions in this analysis. This was expected, as the distilled version of MSDNet gives earlier classifiers access to features composed in deeper layers, which ought to flatten any RT differences. The values on the right vertical axis indicate the mean processing time for each of $K = 5$ auxiliary classifiers. Subplots display grand means for each condition. Best viewed in colour.

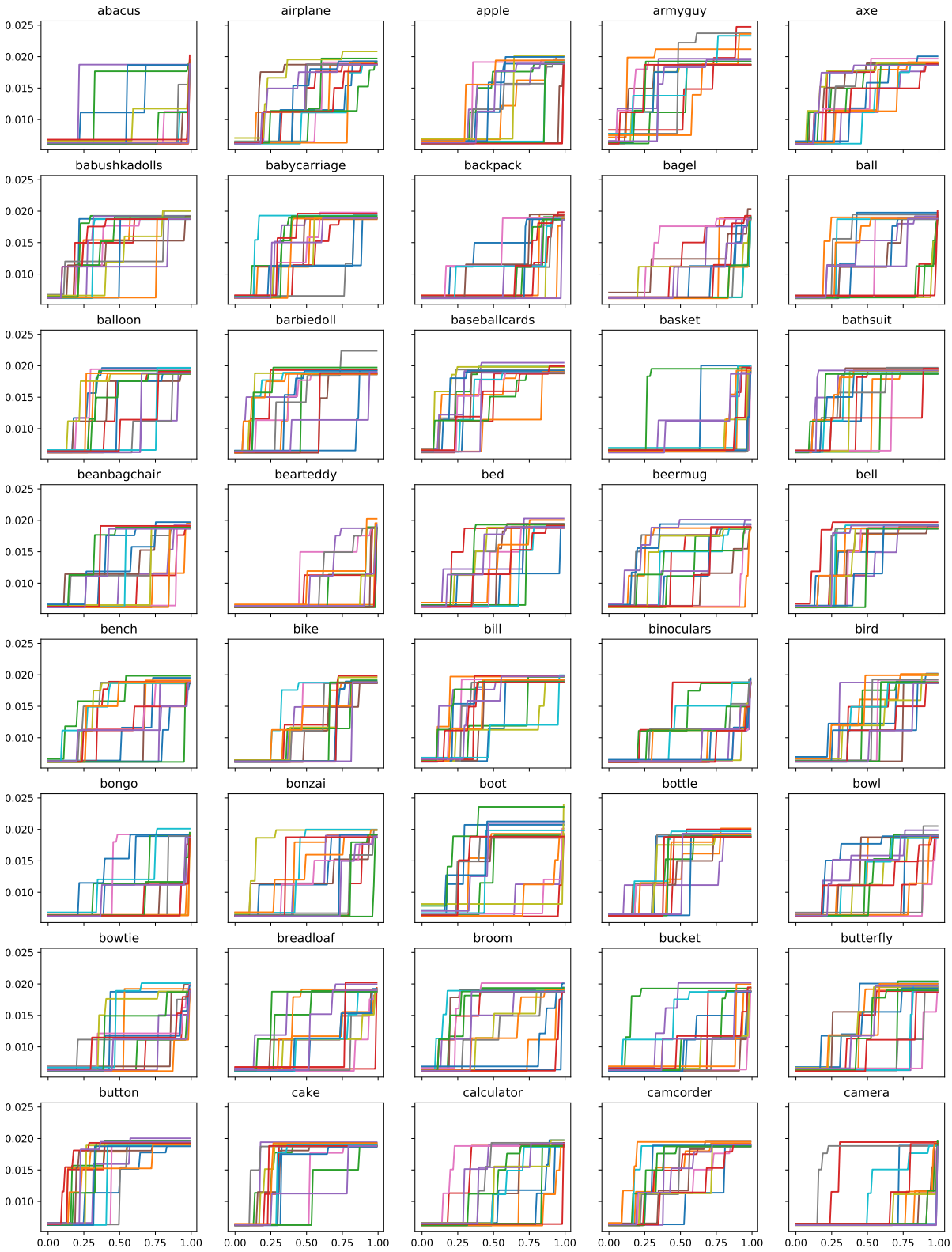


FIGURE C6 Classes 1-40 of the Massive Memory dataset. Note there are some classes that do not overlap with ImageNet. For our purposes, we are mostly interested in seeing how RT profiles for exemplars with similar features (regardless of dataset) proceed through time. MM is handy because objects are pictured on a featureless background, so any commonalities between RT profiles can be attributed to features unique to that input.

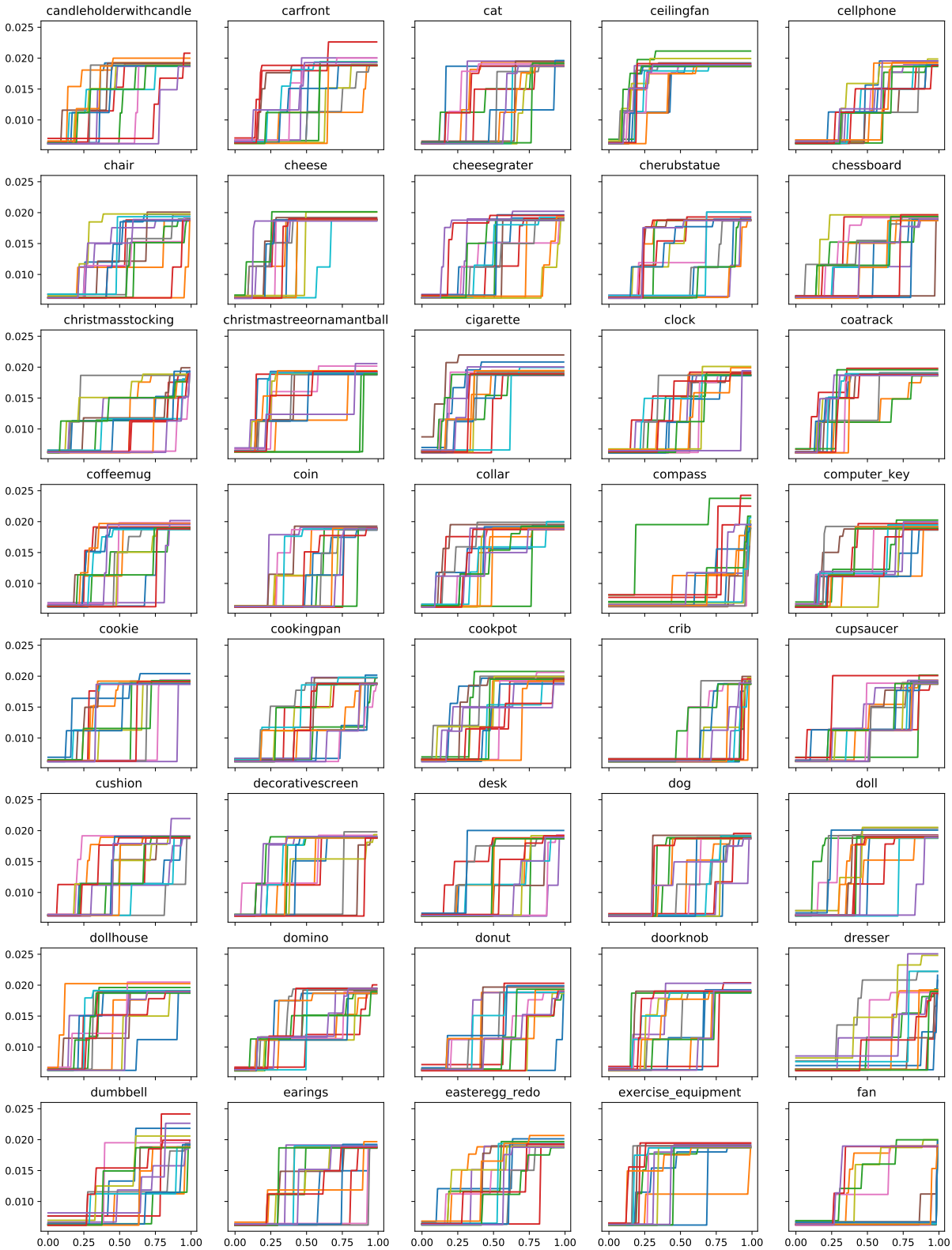


FIGURE C7 Classes 41-80 of the Massive Memory dataset. Note there are some classes that do not overlap with ImageNet. For our purposes, we are mostly interested in seeing how RT profiles for exemplars with similar features (regardless of dataset) proceed through time. MM is handy because objects are pictured on a featureless background, so any commonalities between RT profiles can be attributed to features unique to that input.

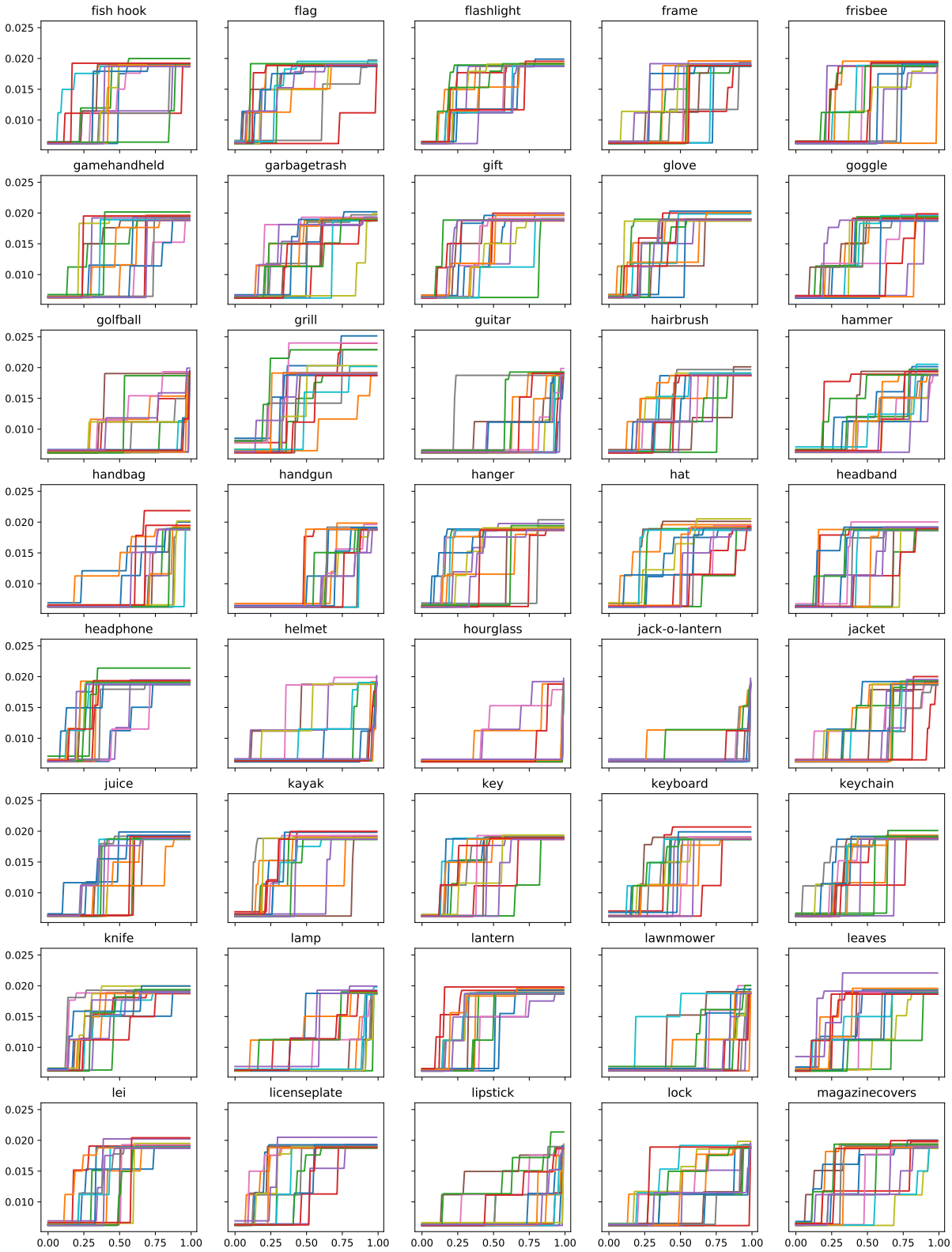


FIGURE C8 Classes 81-120 of the Massive Memory dataset. Note there are some classes that do not overlap with ImageNet. For our purposes, we are mostly interested in seeing how RT profiles for exemplars with similar features (regardless of dataset) proceed through time. MM is handy because objects are pictured on a featureless background, so any commonalities between RT profiles can be attributed to features unique to that input.

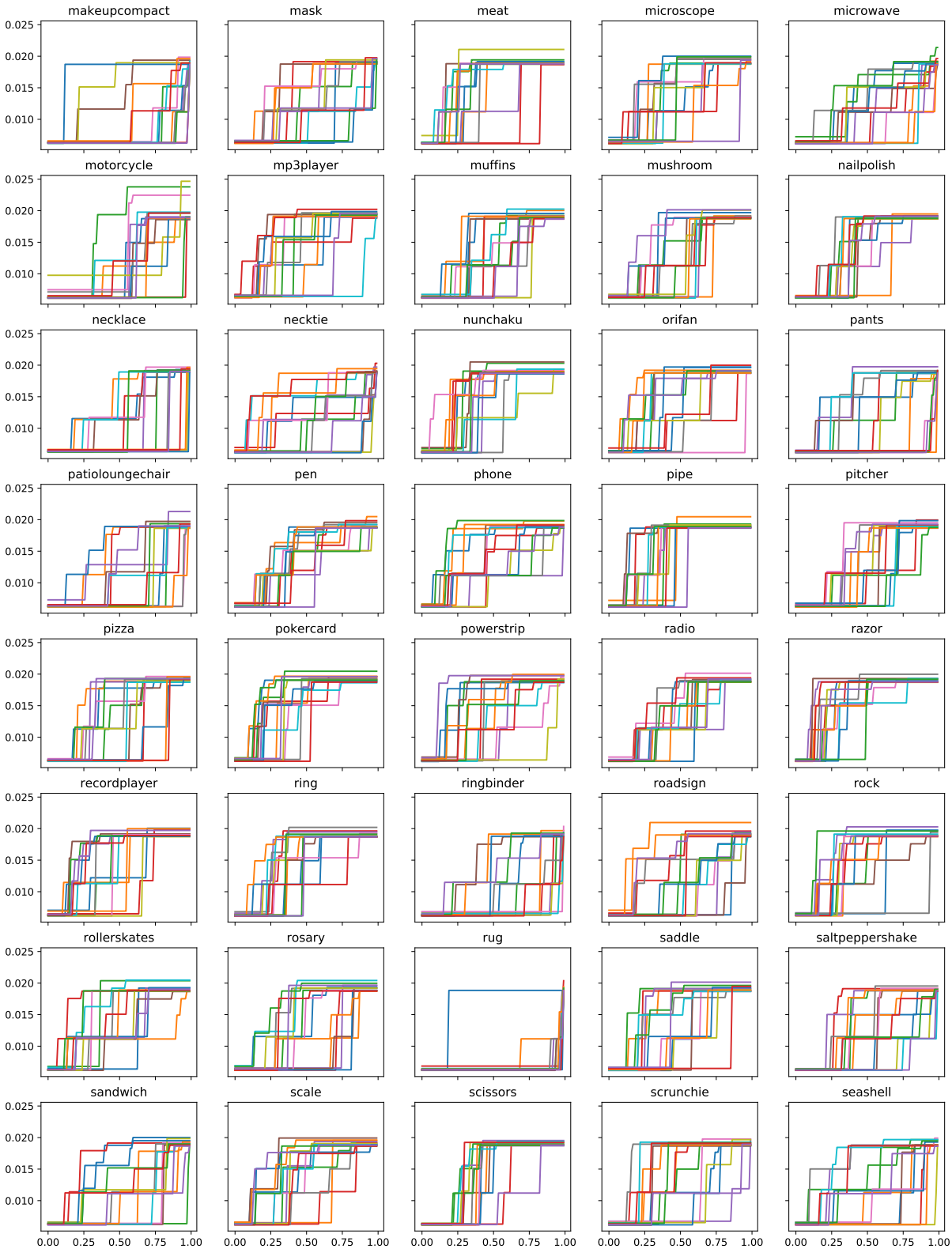


FIGURE C9 Classes 121-160 of the Massive Memory dataset. Note there are some classes that do not overlap with ImageNet. For our purposes, we are mostly interested in seeing how RT profiles for exemplars with similar features (regardless of dataset) proceed through time. MM is handy because objects are pictured on a featureless background, so any commonalities between RT profiles can be attributed to features unique to that input.

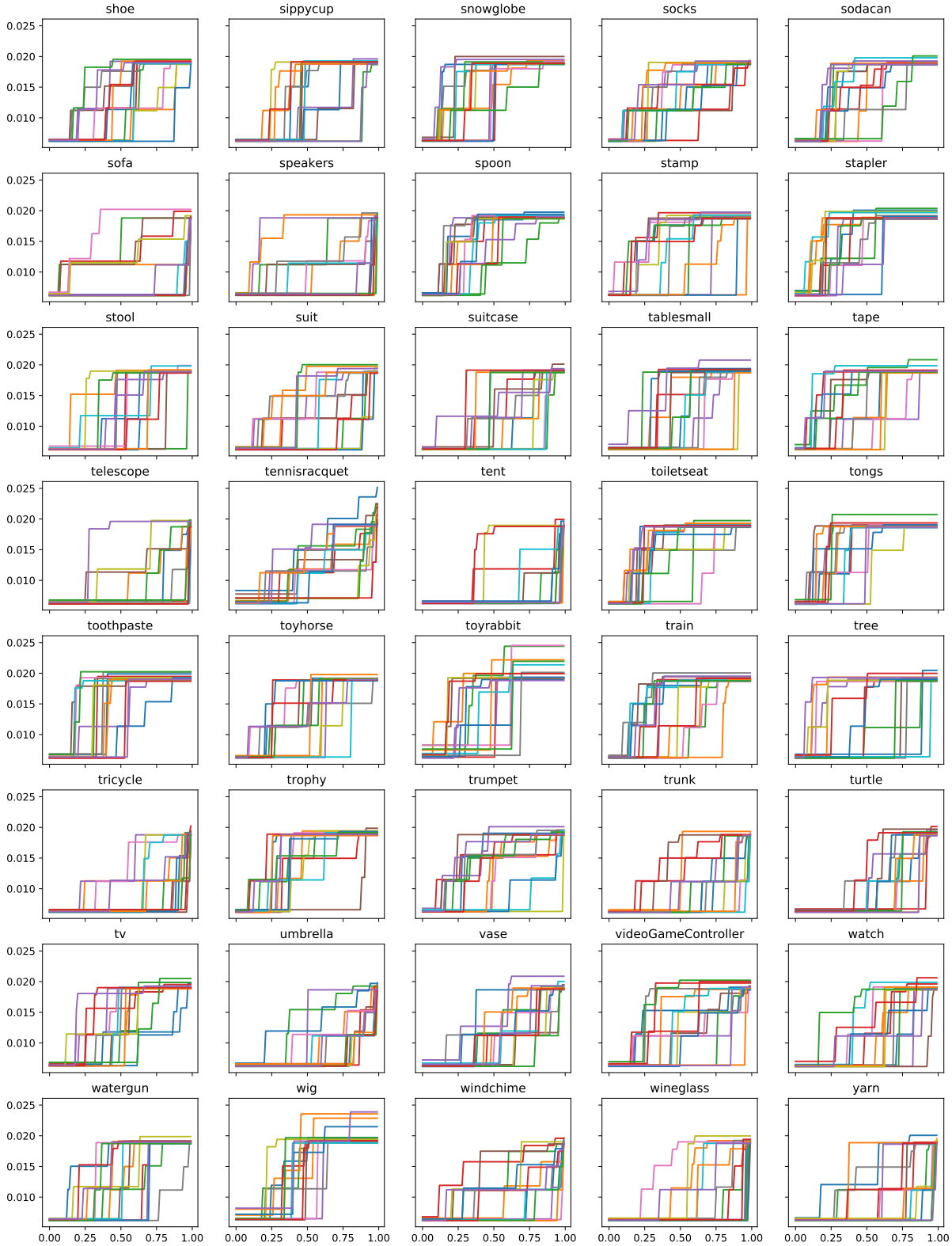
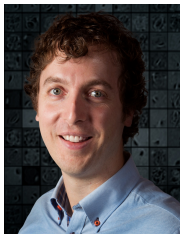


FIGURE C10 Classes 161-200 of the Massive Memory dataset. Note there are some classes that do not overlap with ImageNet. For our purposes, we are mostly interested in seeing how RT profiles for exemplars with similar features (regardless of dataset) proceed through time. MM is handy because objects are pictured on a featureless background, so any commonalities between RT profiles can be attributed to features unique to that input.

AUTHOR BIOGRAPHY

J. Eric T. Taylor Eric Taylor is a postdoctoral fellow at the Vector Institute for AI and the University of Guelph in Ontario, Canada. A cognitive psychologist by training, Eric has extensive fellowship and industry experience. He applies his experience in visual perception, attention, and decision making to new work in artificial cognition, the study of explaining AI via behavioural experiments.

Shashank Shekhar is a master's student at the College of Engineering and Physical Sciences, University of Guelph, Ontario, Canada. He works as a research assistant with the Machine Learning Research Group and is also a recipient of the Vector Institute research grant and Scholarship in Artificial Intelligence. He received his bachelor's in Electronics And Communication Engineering from Indian Institute of Technology (ISM) Dhanbad, India and worked as a project assistant at the Visual Computing Lab, Department of Computational and Data Sciences, Indian Institute of Science. His research focuses on higher-order computer vision, particularly reasoning and explainability.



Graham W. Taylor is a Canada Research Chair and Associate Professor with the University of Guelph, Ontario, Canada, and a Canada CIFAR AI Chair at the Vector Institute for Artificial Intelligence. He received his bachelor's and master's degrees in applied science from the University of Waterloo, Canada, in 2003 and 2004, respectively. He received his Ph.D. degree in computer science from the University of Toronto, Canada, in 2009. He did post-doctoral research at New York University (NYU), New York, NY, USA from 2009–2012. In 2016 he was named a CIFAR Azrieli Global Scholar in Learning in Machines and Brains. In 2018, he was named one of Canada's Top 40 under 40 and Google Visiting Faculty. Through his research, he aims to discover new algorithms and architectures for deep learning.