Short Paper

# Towards Automated Concept-based Decision Tree Explanations for CNNs

Radwa Elshawi
University of Tartu
Tartu, Estonia
radwa.elshawi@ut.ee

Youssef Sherif
University of Tartu
Tartu, Estonia
Youssef.sherif@ut.ee

Sherif Sakr
University of Tartu
Tartu, Estonia
sherif.sakr@ut.ee

## ABSTRACT

Currently, deep learning models have been widely used in different application domains due to their notable performance. Explaining the decisions made by deep learning models is important for end-users to enable them to comprehend and diagnose the trustworthiness of the model. Most of the current interpretability techniques provide explanations in the form of importance score for the input pixels or features. However, summarizing such importance scores for input features to provide human-interpretable explanations is challenging. To this end, we propose *Automated Concept-based Decision Tree Explanations* (ACDTE), a novel local explanation framework that provides human-understandable and concept-based explanations for classification networks. Our framework provides end users with the flexibility of customizing the explanations by allowing users to provide the dataset in which visual human-understandable concepts are automatically extracted. Then, such concepts are interpreted through a shallow decision tree that includes concepts that are deemed important to the model in predicting the decision of specific instance. In addition, ACDTE generates counterfactual explanations, suggesting the the minimum changes in the instance's concept-based explanation that lead to a different prediction. Our experiments demonstrate that such a shallow decision tree is faithful to the original neural network at low tree depth. The human interpretability of the explanations provided from our framework is evaluated through humans experiments, showing that our framework generates faithful and interpretable explanations.

## 1 INTRODUCTION

Since deep learning (DL) models have been achieving remarkable success over the last years in different application domains [1, 3], gaining insights into such models' predictions has received great attention over the last few years and in some cases, there is also a legal requirement to do so [7]. Among the various DL models, convolution neural networks (CNN) achieve remarkable performance in different computer vision tasks including self-driving cars and medical diagnoses. A main drawback for DL models, that prevents their wide adoption in critical domains, is their inscrutable nature of their prediction process that makes them black-boxes. Explaining the behaviour of DL models enables humans to understand the model behaviour, and hence, can increase their trust in the model if the decisions made by the model appear reasonable to humans.

There is no agreement among researchers about what would constitute a satisfactory explanation [13]. However, recent studies over 250 papers have concluded that explanations are counterfactual [12, 13]. Techniques for explaining DL models can be broadly partitioned into two main approaches. The first approach is to identify the evidence that the network uses to make a specific prediction by creating a heatmap that identifies the main parts of the image, which are salient to the prediction [16, 19, 21]. The second approach focuses on providing explanations in the form of human-understandable concepts [5, 6, 20]. Instead of assigning an importance score for each pixel or input feature, the explanation comes in the form of important human-understandable concepts that contribute toward the prediction. Understanding how concepts affect a particular model prediction may reveal potential unwanted bias learned by the model.

In this paper, we describe a framework called Automated Concept-based Decision Tree Explanations (ACDTE) to automatically identify high-level human-understandable concepts which are important for the machine learning model for predicting the decision of a specific instance by aggregating related local image segments (concepts) across diverse data and then decompose the evidence for a prediction for image classification into such concepts through an interpretable shallow decision tree. The explanation provided by the ACDTE framework is expressive and provides not only succinct evidence why a particular image has been assigned to a particular class, but also counterfactuals suggesting what is the least number of concepts needed to be changed are, in an instance's explanation, to change the predicted outcome. We summarize our contributions as follows: 1) A novel local explanation framework to provide automatically extracted concept-based explanations for CNNs in the form of important concepts for the prediction of specific instance presented as a shallow interpretable decision tree that is faithful to the black-box model, 2) A counterfactual explanation, suggesting the changes in the important concepts for the prediction of a specific image that lead to a different outcome, 3) Evaluation of the faithfulness of the explanations provided by ACDTE to the black-box model and the quality of the provided explanations. For ensuring repeatability as one of the main targets of this work, we provide access to the source codes and the detailed results for the experiments of our study[1].

The remainder of this paper is organized as follows. In Section 2, we present our proposed technique, ACDTE. We present a detailed experimental evaluation for our proposed techniques in Section 3 before we finally conclude the paper in Section 4

## 2 METHODS

In the following, we present ACDTE which is a local explanation technique that explains the prediction of a particular image. ACDTE takes a trained classifier, an image to be explained, and a set of images from user-specified dataset as input. It then extracts concepts presented in these images and interpret these concepts through a shallow decision tree that identifies the main concepts



[1] https://github.com/DataSystemsGroupUT/ACDTE

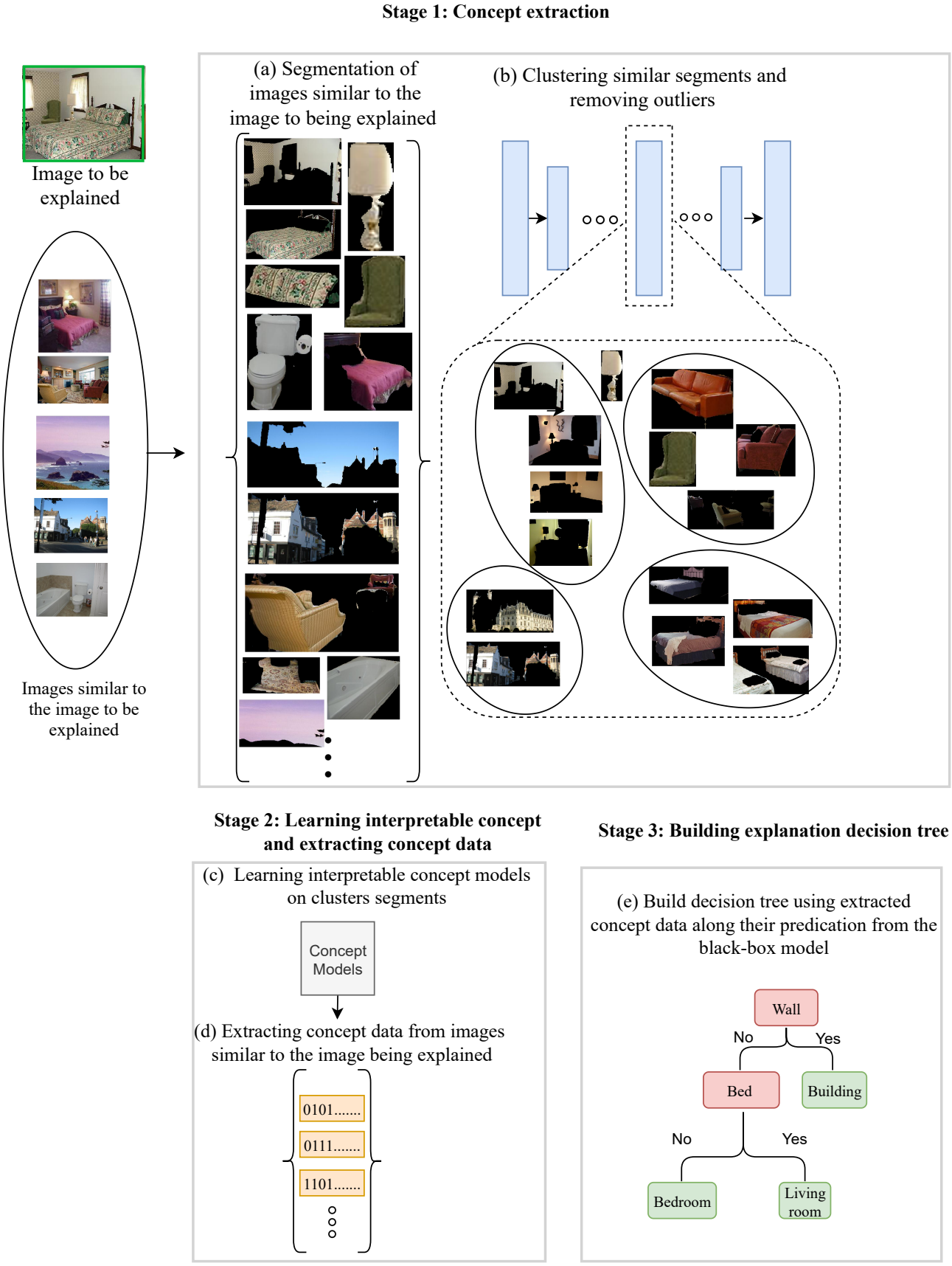


**Figure 1: ACDTE pipeline (a) Extract a set of similar images to the image to be explained either from the main task dataset or related dataset. Each image in the selected images is segmented. (b) Segments are clustered in the activation space and outliers are removed to form coherent clusters that represent concepts. (c) Training a linear model for each concept to act as a concept detector. (d) For each image in the activation space, use concepts detectors to form a binary feature vector. (e) Feature vectors along with the prediction of the target network are used to train a shallow decision tree. The decision tree provides a natural explanation for the contributing concepts for the prediction, in addition to counterfactual explanation.**

that have deemed important for the prediction of the image being explained, in addition to the minimum number of concepts that need to be changed to alter the prediction of the image to be explained. Figure 1 summarizes the general pipeline for the proposed framework consists of three main phases concept extraction, learning interpretable concept models and extracting concept data, and building explanation decision tree in which explanation based on both decision and counterfactuals, is extracted.

## 2.1 Phase 1: Concept Extraction

Given a pretrained image classification model $m$, and the image to be explained $I$, our framework provides end users with the flexibility of extracting concepts either from the dataset used in the classification task or from a related dataset to the main task dataset. To extract concepts, we choose the top $k$ images similar to $I$, denoted $S$. Similarity between $I$ and the set of provided images is defined to be the Euclidian distance between their corresponding activation maps obtained from an intermediate layer from $m$. In this paper, we use a constant value for $K = 100$, leaving the exploration of different values to future work. To extract concept data, each of the images in $S$ is segmented using semantic image segmentation technique, see Figure 1(a). In order to automate the process of concept extraction, a significant number of studies in literature focused on semantic segmentation algorithms that aim to assign a meaningful class to each pixel [9, 11, 14, 18]. ACDTE uses DeepLabv3+ [2] segmentation technique which has been widely used due to its superior performance on dense datasets (after examining several segmentation techniques). To ensure meaningfulness of the extracted concepts, we cluster segments into a number of clusters such that segments of the same cluster represent a particular concept. In order to automate the process of clustering segments, we define the similarity between segments to be the euclidian distance between their corresponding activation maps obtained from the intermediate layer of model $m$. Each segment was resized to the original size of $m$. All segments were then passed through $m$ to obtain their layer presentations. All segments are then clustered using K-means clustering algorithm [10], see Figure 1(b). To ensure meaningfulness of the extracted concepts, we exclude the following two types of clusters: 1) Clusters that have segments that only coming from a single image or a very few number of images. 2) Clusters with segments less than $N$ segments. In this work, we use a constant value for $N$ equals $0.4\sqrt{n_c}$, where $n_c$ is the number of segments in cluster $c$, leaving the exploration of different values for $N$ to future work. The main problem with clusters of few segments is that the concepts they present are uncommon in the neighborhood of the image being explained. For example, bed segments are present in almost every bedroom image and therefore, are expected to form a coherent cluster while lamp segments are presented in very few bedroom images and hence lamp cluster should be removed. The output of this phase is the final set of clusters representing the learnt concepts denoted $C = \{c_1, ...c_n\}$, where $n$ is the number of clusters after the exclusion criteria.

## 2.2 Phase 2: Learning Interpretable Concept Models and Extracting Concept Data

For each segment $x \in c$, the hidden layer activations $a = m_l(x)$ at layer $l$ are extracted and stored along its corresponding concept label. For each candidate concept $c \in C$, we train a logistic binary classifier $h_c$ to detect the presence of concept $c$, see Figure 1(c). Training each concept $c$ is done on dataset $D_c$, which is mix of segments balancing the presence and absence of concept $c$. We define $D_c = D_c^+ \cup D_c^-$, where $D_c^+ = \{(m_l(x^1), y_c^1), ...., (m_l(x^{|c|}), y_c^{|c|})|_{y_c=1}\}$ and $D_c^- = \{(m_l(x^1), y_c^1), ...., (m_l(x^{|c|}), y_c^{|c|})|_{y_c=0}\}$, where $y \in \{0, 1\}$ indicates the absence or the presence of concept $c$ in a segment. Negative examples $D_c^-$ for each concept $c$ are selected randomly from other cluster concepts such that the number of examples in $D_c^+$ and $D_c^-$ are equal. We use these concept classifiers for each image $s \in S$ to create a binary vector $v = (r_1, r_2, ..., r_n)$ representing the presence or absence of each concept $c \in C$ in $s$, where $r_i = h_{c_i}(s)$, $r_i \in \{0, 1\}$. For each image $s \in S$, we store its class prediction from model $m$ along with its binary concept vector $v$ for training a decision tree, see Figure 1(d).

## 2.3 Phase 3: Building Concept Decision Tree

Concept vector $v$ predicted for each image $s \in S$ along with the corresponding prediction $m(s)$ are used to train a decision tree $T$ which is intended to mimic the behavior of $m$ locally in the $S$ neighborhood, see Figure 1(e). We use the default implementation of decision tree from *scikit-learn* [15]. The ACDTE approach considers decision tree classifier due to its interpretable nature that allows concept rules to be derived from a root-leaf path in the decision tree, in addition to counterfactuals that can be extracted by symbolic reasoning over a decision tree. Increasing the depth of a decision tree increases the prediction accuracy which leads to less interpretable results as the number of nodes increases exponentially with depth. Thus, a shallow decision is favourable as it is more comprehensible by humans. In this work, we use a fixed depth leaving the exploration of dynamic depth to future work. In order to guarantee fast and easy search for counterfactuals, we consider all possible paths in the decision tree leading to a decision that is not equal to the decision of $I$. Among all these paths, we only consider the one with the minimum number of spilt conditions that are not satisfied by instance $I$. As an example, consider the decision tree in Figure 2 explaining the prediction from ResNet50 pretrained on places dataset [22] of an image as a coast. The concepts used in building the decision tree is based on selecting the top 100 images from a random selection of 1000 images from the ADE20k dataset [23]. The left branch of the tree indicates the presence of a concept while the right branch indicates the absence of that concept. The tree gives insights into the main human-understandable concepts from ADE20K dataset that appear important for ResNet50 in predicting the coast image. The decision tree provides a natural explanation for each path. It is clear from the explanation tree that the image has been predicted as a coast because of the existence of the concepts 'mountain' and 'sea'. As a further output, ACDTE computes a counterfactual; we have two counterfactual paths in the decision tree shown in Figure 2. The first one is the presence of 'mountain', absence of 'sea', presence of 'tree' that leads to the prediction of class 'snowy mountain', while the second is the presence of 'mountain', absence of 'sea', absence of 'tree' that leads to the prediction of class 'highway', as shown in Figure 2. Figure 3 shows sample segments of concepts along the explanation path of the coast image shown in Figure 2.

# 3 EXPERIMENTS AND RESULTS

In this section, we evaluate the meaningfulness of the explanations provided by our framework. In addition, we evaluate the faithfulness of the proposed framework to the black-box model.

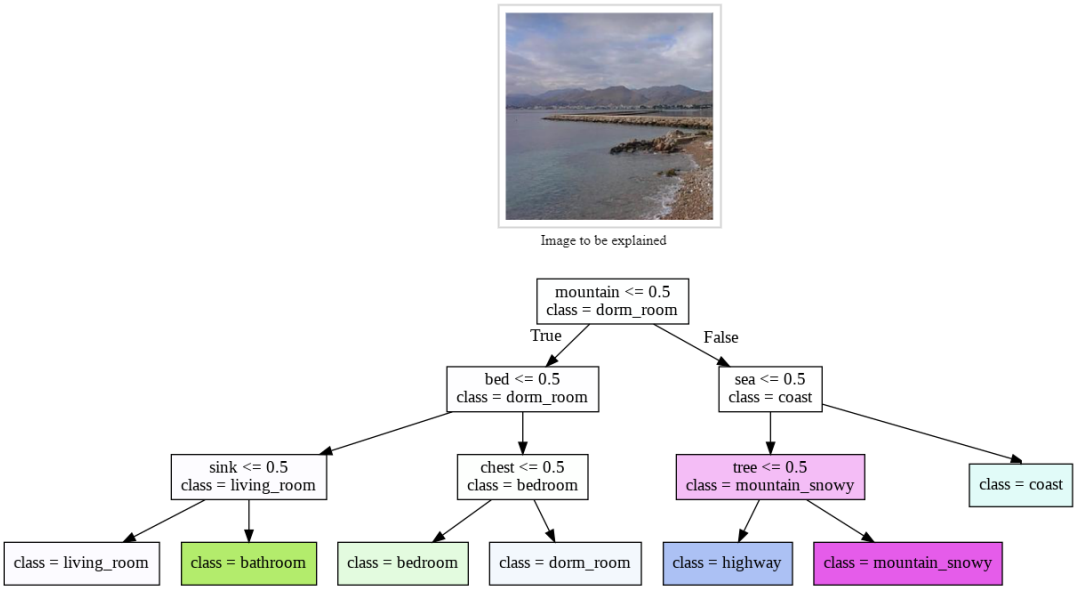


**Figure 2: Shallow concept-based explanation decision tree of depth 4 explaining the prediction of coast image.**


**Figure 3: Sample segments of concepts along the explanation path of the coast image shown in Figure 2. Text below each group of images describes its original class of the extracted concepts.**

## 3.1 Experiment Setup

As an experimental example, we use ACDTE to explain the predictions of the widely-used Resnet50 that has been pre-trained on the places dataset. We select a subset of 30 classes out of the 365 classes from places dataset. We experimented extracting the concepts from ADE20K dataset. More specifically, to explain the prediction of an instance from places dataset, we randomly select 1000 images from ADE20K dataset and extract concepts from the nearest 100 images. To evaluate the performance of the ACDTE, we randomly select 1000 images, denoted $X$, from the 30 selected classes of the places dataset.

## 3.2 Are ACDTE Explanations Faithful to the Black-box Model?

We consider the following metrics in evaluating how well the decision tree inferred by ACDTE and the explanations returned mimic the black-box model.

- **fidelity** $\in [0, 1]$: It compares the prediction of the decision tree $T$ and the black-box model $m$ on the set of images $S$ used to train the decision tree [4].
- **hit** $\in \{0, 1\}$: It compares the prediction of the decision tree $c$ and the black-box model $m$ on the instance to be explained $I$ [8]. It returns 1 if $m(I) = T(I)$, and 0 otherwise.

We measure the fidelity by F1-measure [17] and report the aggregated values of the F1 measure across all instances in $X$ at tree depth of 5, 10, 15 and 20, see Figure 4(a). We report hit by averaging its values across the instances in $X$ at tree depth of 5, 10, 15 and 20, see Figure 4(b). The results show that fidelity and hit increases as the tree depth increases, however tree depth of 10 is able to achieve reasonable fidelity of 0.87 and hit of 0.91.

## 3.3 Examining the Significance of the Extracted Concepts from ACDTE

To confirm the importance of the formed concepts of ACDTE, we run ACDTE on each of the images in $X$ and return the set of clusters obtained from the concept extraction phase. We rank the returned clusters for each image in $X$ according to their compactness that is captured by calculating the average distance between cluster center and each point in the same cluster. The smaller the average distance indicates that the cluster is tightly formed and shows a motion coherent view. The intuition behind that ranking is that compact clusters most likely represent a concept that is frequently present in the neighbourhood of the image to be explained, and hence, have a significant role in forming the decision boundary between classes. For each image in $X$, we build different decision trees based on excluding the top $k$ concepts obtained from the concept extraction phase, where $k$=0, 2, 5, 8 and 10. Figure 5 shows the prediction accuracies on the set of images used to train the decision tree when removing the most important $k$ concepts aggregated across all the instances in $X$. The results show that accuracy decreases significantly from 92.4% to 75.3% when removing the 10 most important concepts which reflects the variable significance of the automatically extracted concepts.

## 3.4 Concept Classifier Prediction Performance

Concept models performance vary across the different layers of the main task model (ResNet50). In order to identify the best

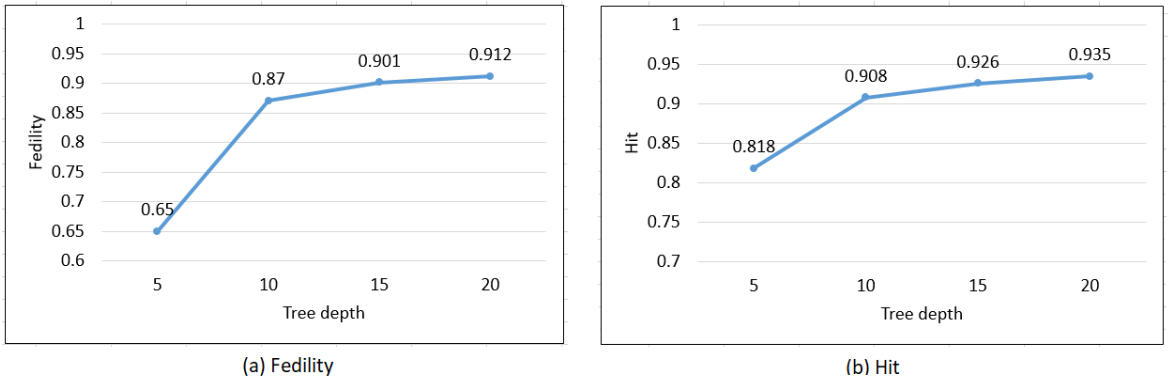


**Figure 4: Fidelity and hit at different tree depth**

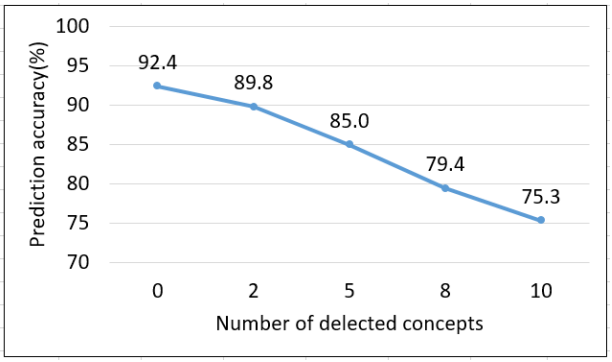


**Figure 5: Prediction accuracy of decision tree as removing the top $k$ important concepts aggregated across all the instances in $X$.**

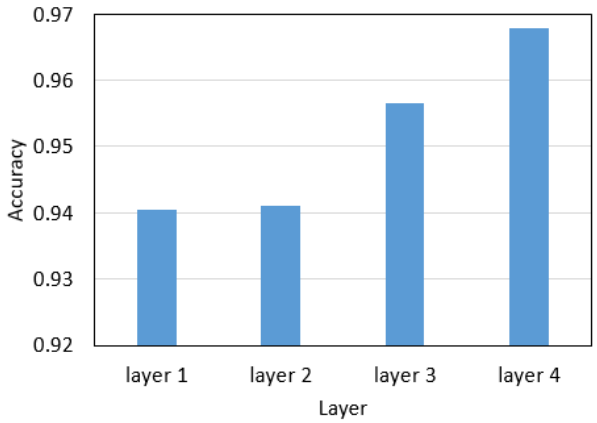


**Figure 6: Average accuracy of all concept classifiers trained for the main layers of ResNet50**

layer to extract feature vectors used to train concept classifiers, we compare the average accuracy of the concept models built on vectors extracted from the major layers of Resnet50 and report this average accuracy averaged over all instances in $X$. Major layers refer to the conv2 x (layer1), conv3 x (layer 2), conv4 x (layer 3), and conv5 x (layer 4) blocksections of sublayers of Resnet50. Figure 6 shows that all layers have high average accuracy and the deeper the extraction layer, the higher the accuracy. Figure 6 shows that the average classifiers accuracy was the highest at the fourth layer, achieving an accuracy of 0.97.

## 3.5 Decision Tree Performance

Figure 7 shows how the accuracy of the decision tree obtained from ACDTE responds to the changes in the maximum tree depth and the layer from which deep features are extracted, to train the concept models. We incrementally increase the depth of the decision tree obtained from ACDTE for each instance in $X$ and change the layer in which features from ResNet are extracted to train the concept models. Then, we measure the prediction accuracy of the instances used to train the decision tree and report this accuracy averaged over all instances in $X$. Result show that the accuracy improves significantly as more concepts are added and then slightly flatten out as depth increases beyond 15. The results also demonstrates that layer 4 of ResNet50 achieves the best performance in terms of the decision tree prediction accuracy.

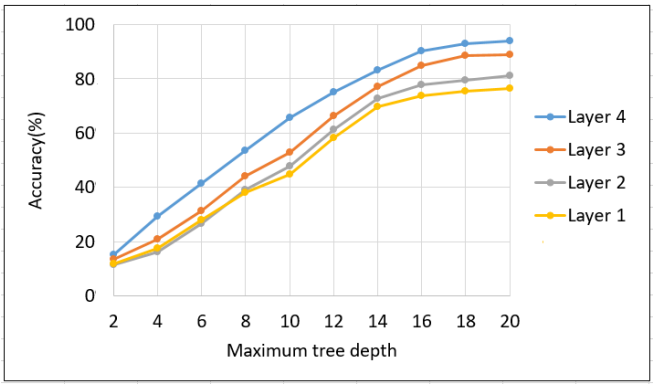


**Figure 7: Decision tree accuracy vs. decision tree depth**

## 3.6 Human Evaluation of the Visual Explanations

To measure the meaningfulness of the extracted concepts, we randomly select 50 instances from $X$ and get the concepts used in their explanations. We ask 30 human participants to identify that segments belong to a concept versus a random set of segments. The evaluation interface is shown in Figure 8. Results show that 87% of participants choose the concept segments. To measure the significance of the important concepts extracted from the ACDTE,

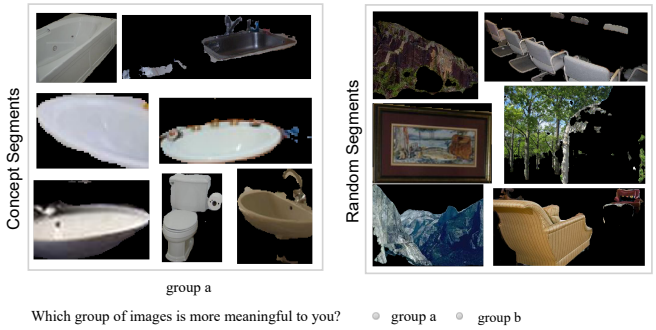


**Figure 8: Human evaluation interface for identifying meaningful concepts**

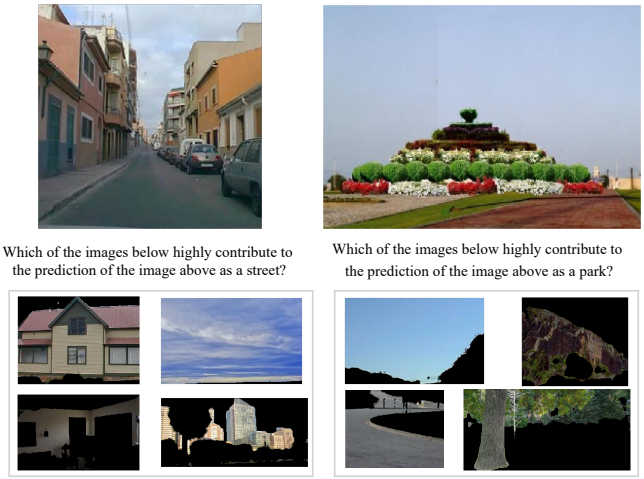


**Figure 9: Sample examples of human experiments for choosing the most contributing concept to their predictions**

we ask the 30 participants to select the most meaningful concept that contributes to a particular prediction made by ResNet50 for 30 different images. In each task, participants are shown the image to be explained along with its prediction and four concepts in which one of them represents the top concept identified by ACDET for explaining this image and the other three concepts are randomly chosen. Participants are asked to select the most meaningful concept that contribute to the prediction. Figure 9 shows two sample images along with four different concepts in which participants are asked to choose the most contributing concept for the prediction of these images. On average, 85% of the participants chose the concept obtained by ACDTE as the most important concept.

## 4 CONCLUSION

We introduced ACDTE, a post-training local explanation technique that automatically extract groups of input features from images similar to the images to be explained and group these features into high-level human-understandable concepts. We verified the meaningfulness and coherence of these concepts through human experiments and further validated that these concepts carry some signals indicating to the correct prediction class for the instance to be explained. Representing these concepts in a shallow decision tree allows users to infer which concepts are significant in the prediction of the image to be explained. A future direction of automated concept-based explanation is to consider other types of data such as texts.

## ACKNOWLEDGMENT

The work of Sherif Sakr and Youssef Sherif is funded by the European Regional Development Funds via the Mobilitas Plus programme (grant MOBTT75). The work of Radwa Elshawi is funded by the European Regional Development Funds via the Mobilitas Plus programme (MOBJD341).

## REFERENCES

[1] Javed Ashraf, Asim D Bakhshi, Nour Moustafa, Hasnat Khurshid, Abdullah Javed, and Amin Beheshti. 2020. Novel Deep Learning-Enabled LSTM Autoencoder Architecture for Discovering Anomalous Events From Intelligent Transportation Systems. *IEEE Transactions on Intelligent Transportation Systems* (2020).

[2] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*. 801–818.

[3] Tobias Domhan, Jost Tobias Springenberg, and Frank Hutter. 2015. Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

[4] Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608* (2017).

[5] Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. 2019. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems*. 9273–9282.

[6] Abel Gonzalez-Garcia, Davide Modolo, and Vittorio Ferrari. 2018. Do semantic parts emerge in convolutional neural networks? *International Journal of Computer Vision* 126, 5 (2018), 476–494.

[7] Bryce Goodman and Seth Flaxman. 2017. European Union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine* 38, 3 (2017), 50–57.

[8] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Dino Pedreschi, Franco Turini, and Fosca Giannotti. 2018. Local rule-based explanations of black box decision systems. *arXiv preprint arXiv:1805.10820* (2018).

[9] Wei Liu, Andrew Rabinovich, and Alexander C Berg. 2015. Parsenet: Looking wider to see better. *arXiv preprint arXiv:1506.04579* (2015).

[10] Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE transactions on information theory* 28, 2 (1982), 129–137.

[11] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3431–3440.

[12] Tim Miller. 2018. Contrastive explanation: A structural-model approach. *arXiv preprint arXiv:1811.03163* (2018).

[13] Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence* 267 (2019), 1–38.

[14] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. 2015. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*. 1520–1528.

[15] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12, Oct (2011), 2825–2830.

[16] Ramprasaath R Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. 2016. Grad-CAM: Why did you say that? *arXiv preprint arXiv:1611.07450* (2016).

[17] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. 2016. *Introduction to data mining*. Pearson Education India.

[18] Xing Wei, Qingxiong Yang, Yihong Gong, Narendra Ahuja, and Ming-Hsuan Yang. 2018. Superpixel hierarchy. *IEEE Transactions on Image Processing* 27, 10 (2018), 4838–4849.

[19] Wencan Zhang, Mariella Dimiccoli, and Brian Y Lim. 2020. Debiased-CAM for bias-agnostic faithful visual explanations of deep convolutional networks. *arXiv preprint arXiv:2012.05567* (2020).

[20] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2014. Object detectors emerge in deep scene cnns. *arXiv preprint arXiv:1412.6856* (2014).

[21] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2921–2929.

[22] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Antonio Torralba, and Aude Oliva. 2016. Places: An image database for deep scene understanding. *arXiv preprint arXiv:1610.02055* (2016).

[23] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. 2017. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 633–641.