Teaching Categories to Human Learners with Visual Explanations

Oisin Mac Aodha   Shihan Su   Yuxin Chen   Pietro Perona   Yisong Yue
California Institute of Technology

Abstract

We study the problem of computer-assisted teaching with explanations. Conventional approaches for machine teaching typically only provide feedback at the instance level e.g., the category or label of the instance. However, it is intuitive that clear explanations from a knowledgeable teacher can significantly improve a student's ability to learn a new concept. To address these existing limitations, we propose a teaching framework that provides interpretable explanations as feedback and models how the learner incorporates this additional information. In the case of images, we show that we can automatically generate explanations that highlight the parts of the image that are responsible for the class label. Experiments on human learners illustrate that, on average, participants achieve better test set performance on challenging categorization tasks when taught with our interpretable approach compared to existing methods.

1. Introduction

Computer-assisted teaching offers the promise of personalized curricula that are tailored to the ability level and interests of every individual. Providing open access to the kinds of high-quality teaching that is currently only available to a small percentage of the world's population has the potential to transform education. To date, subject areas such as mathematics [18, 5] and language learning [34] have benefited from automated teaching systems. However, the problem of how to best teach visual expertise in fine-grained domains such as medical diagnosis and species identification is comparatively less well explored.

In addition to the benefits to human learners from better teaching methods, automated systems could also take advantage of these improvements. Access to expert time is often limited, and as a result, there is a need for better techniques to train crowd workers for image annotation tasks. Once the workers have effectively learned the task they can provide higher quality labeled training data. This 'closing of the loop' will enable us to take advantage of humans' ability to both generalize across different domains and cope with other nuisance factors such as pose and lighting changes.
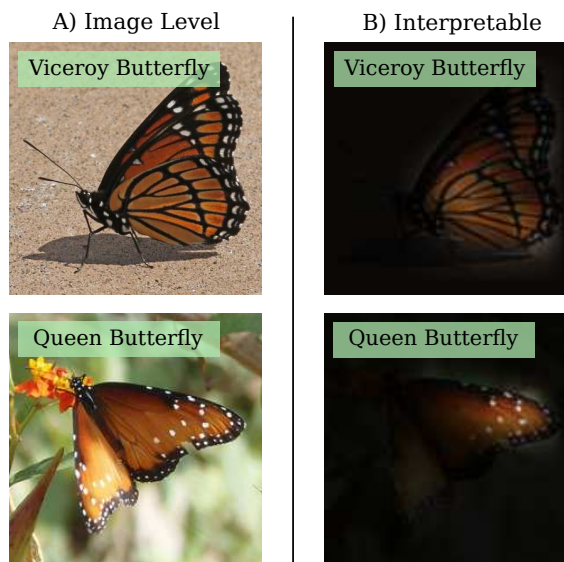

Figure 1. A) The majority of existing machine teaching algorithms for visual categories only give feedback to the learner in the form of the ground truth class label. B) It is much more informative to display the discriminative regions to help them to determine the categories present in the images. Here, we see explanations from our system highlighting the blank band on the Viceroy Butterfly and the white spots on the Queen Butterfly's wings. These are field markings commonly used to identify both species.

Existing approaches to teaching visual knowledge typically pose the problem as choosing the most informative subset of images to show from a much larger set of possible options. One of the major limitations of this existing work is that they only give very limited feedback to students in the form of the class label, *e.g.* [31, 16]. In Fig. 1 A) we see an example of this label only feedback. Providing only the ground truth class label is a very limited amount of feedback compared to the rich explanations that one may receive from a human teacher. A human teacher would likely teach the student the specific parts and attributes in the image that are most discriminative for that particular class, Fig. 1 B).

Our hypothesis is that students that are taught with interpretable feedback will learn more effectively than those that only receive label feedback. Specifically, we propose a novel teaching algorithm that selects images that are both repre-

sentative of the categories of interest and have explanations that can be easily interpreted. These explanations come in the form of feedback during teaching indicating the parts of the image that are important for successful classification. We show that it is possible to generate these explanations from existing labeled datasets, thus minimizing the need for additional annotations. Through experiments on real human learners, we show that our joint selection of informative images and interpretable explanations results in better student learning and improved generalization at test time.

2. Related Work

Machine Teaching

The goal of machine teaching is the design of algorithms and systems that can teach humans efficiently and automatically. To date, a variety of different approaches have been explored for modeling the teaching of students from assuming perfect learners [12, 40, 21], heuristic-based approaches [2], Bayesian models [7, 11], recurrent neural networks [23], and reinforcement learning based methods [25, 1, 37].

While machine teaching has been successfully deployed in online tutoring systems that feature highly structured knowledge *e.g.* mathematics, the teaching of challenging *visual* concepts to human learners is less explored. [31] teach binary visual classification tasks by modeling the student as stochastically switching between a set of different hypotheses during learning. Their model attempts to select the set of teaching examples offline that will best discount the incorrect hypotheses and guide the student towards the ground truth classification function. [16] propose an interactive approach, where the choice of future images to show is based on the individual's past responses. However, they must update the model's parameters online for each user making it computationally difficult to scale to large numbers of simultaneous users in real-world settings.

The major limitation of these existing approaches is that the feedback they provide to the student is not fully informative. In both [31] and [16] a student is shown a sequence of images and asked to estimate what object category from a finite list they believe to be present in each image. After they respond they are simply told what the correct answer is. They are not informed of the parts in the image that are discriminative for identifying that particular object, thus making the learning problem artificially hard for the student. To overcome this limitation, we propose a novel teaching algorithm that selects both images and provides interpretable explanations resulting in more understandable and efficient learning for the student. Complementary to our work, [6] recently introduced an explanation based teaching algorithm for binary tasks. Explanations are provided via pre-existing semantically meaningful features, but the interpretability of a given explanation is not modeled.

Interpretable Models

Using clear and understandable instructional material can dramatically improve a student's ability to learn a new concept. It has been shown that highlighting informative regions on an image can help improve novice classification performance by guiding the student's attention [13, 27].

In another example, when a human teacher is unavailable, the most common way novices learn species identification is by consulting expertly curated field guides. These field guides are typically books or apps that contain descriptive text and example images highlighting important features for classifying different species *e.g.* [22]. Attempts have been made to automate the creation of these guides using highlighted part annotations [3], automatic generation of image specific text descriptions [15], or through gamification [8]. However, in addition to image level class labels, the majority of these approaches require the collection of *expert* annotations in the form of text descriptions, anatomical part locations, or visual attributes which can be expensive and time consuming to obtain for very large image collections [4]. Furthermore, the efficacy of these annotations for teaching visual identification skills has not yet been evaluated on real human subjects. An alternative approach that requires less additional annotations is to learn human interpretable models from the raw data *e.g.* [26, 19]. In the context of computer vision, there is some evidence to suggest that the deep models commonly used for large-scale image classification tasks can be adapted to generate features that are semantically meaningful to humans [39, 38].

Recently, [24] outlined an approach for incorporating additional supervised data from users which they call 'feature feedback'. In addition to class level labels that are typically provided by human annotators when training supervised classifiers, they allow their annotators to provide information about the value of specific feature dimensions. In contrast, our approach instead gives explanations to the *learner* about the importance of different image regions and models how they incorporate this information when updating their belief.

We are concerned with selecting the set of teaching examples with associated interpretable explanations to best teach noisy human learners. While in practice these explanations can be generated with additional time-consuming human annotation, we show that it is possible to extract meaningful explanations using existing image level labels.

3. Interpretable Visual Teaching

In this section, we outline our model for teaching categorization problems to human learners and present an efficient algorithm for selecting interpretable teaching examples. We discuss our approach in the context of the non-adaptive model of [31], but any visual teaching algorithm can benefit from interpretable explanations.

3.1. Problem Setup

We aim to teach a human learner a target classification function h^* that maps from images $\mathcal{X}$ to their corresponding ground truth class labels $\mathcal{Y}$. For instance, a single image $x \in \mathcal{X}$ could be a picture of a bird and the associated label y could be the name of the species. However, we cannot directly impart the parameters of h^* to the human learner, instead we must *teach* them by showing them example images. Given a set of n images, $\mathcal{X} = \{x_1, \ldots, x_n\}$, with associated labels from C classes $\mathcal{Y} = \{y_1, \ldots y_C\}$, our goal is to select an informative subset T, referred to as the teaching set, that will best convey the ground truth, *i.e.* the human teacher's classification function h^*. Acknowledging that some images are more informative than others during learning, we do not want to waste effort teaching these unrepresentative images.

When learning from a human teacher, or even from a textbook, a student not only receives feedback in the form of the correct answer but also an explanation that describes *why* a given answer is correct. In addition to the images and labels, we further assume that we (as the teacher) have access to an explanation, e, for each image x. In the case of images, these explanations could be heatmaps that highlight the informative regions for a particular class in a given image. In non-visual scenarios, this could be a piece of text describing the relationship between x and y.

3.2. Image Only Learner Model

We adopt the stochastic STRICT algorithm of [31] to model how learners adapt to the images shown by the teacher. The model was originally proposed for teaching binary classification functions. Learners are modeled as carrying out a random walk in a finite hypothesis space $\mathcal{H}$. Each element of $\mathcal{H}$ is a function that maps from an image to a score $h : \mathcal{X} \mapsto \mathbb{R}$. In the context of binary classification, where $y \in \{-1, 1\}$, the label predicted by a hypothesis h for image x is $\hat{y}^h = \text{sgn}(h(x))$, and the magnitude indicates the confidence that h has in its prediction. Concretely, an image x may be represented by a feature vector and a hypothesis h could be a linear classifier, $h(x) = w_h^\mathsf{T} x$, with weights w_h. To ensure that it is possible to teach the learner, we assume that $\mathcal{H}$ also contains the ground truth hypothesis h^*.

At the beginning of teaching, the learner randomly picks a hypothesis $h \in \mathcal{H}$ according to the prior distribution $P(h)$. During teaching, she will be presented with a sequence of images along with the correct class label. After receiving a new image, the learner will stick to her current hypothesis if the ground truth label is consistent with the prediction of the hypothesis. Otherwise, she randomly switches to a new $h \in \mathcal{H}$ according to her current posterior over the hypotheses $P(h \mid T)$, where T is the set of teaching images and ground truth labels seen so far. When updating her posterior $P(h \mid T)$ in light of the new information, hypotheses that disagree with the true labels of the images shown by the teacher are

less likely to be selected

$$P(h \mid T) \propto P(h) \prod_{\substack{x_t \in T \\ y_t \neq \hat{y}_t^h}} P(y_t \mid h, x_t), \qquad (1)$$

where $\hat{y}_t^h$ is the class label predicted by hypothesis h for image x_t. We model how much the prediction of hypothesis h agrees with the correct label for image x_t using

$$P(y_t \mid h, x_t) = \frac{1}{1 + \exp(-\alpha h(x_t) y_t)}, \qquad (2)$$

where $\alpha > 0$ represents how consistently a learner responds according to her currently adopted hypothesis. In the extreme when $\alpha = \infty$, hypotheses that are inconsistent with the ground truth label are immediately discarded. This unrealistic setting represents the perfect learner [12].

3.3. Interpretable Feedback

The feedback presented to the learner in the STRICT model is comprised of only the true, ground truth, label of a teaching example. The learner is tasked with learning the mapping between the image and the ground truth label themselves as no other explanations are provided *i.e.*, they must determine which regions or parts in the image are responsible for the given class label. In real-world teaching scenarios, the teacher often has access to much richer information which can be further utilized to accelerate the teaching process. Our approach, EXPLAIN, gives feedback to the learner in the form of *explanations* and models how they incorporate this information when updating their belief.

With a slight abuse of notation, we extend T to denote the set of labeled teaching images along with their explanations. Our updated model of the learner introduces two additional discount terms that account for the interpretability of an explanation and the representativeness of an image

$$P(h \mid T) \propto P(h) \prod_{\substack{x_t \in T \\ y_t \neq \hat{y}_t^h}} P(y_t \mid h, x_t) \prod_{x_t \in T} (E(e_t) D(x_t)). \qquad (3)$$

Modeling Explanations

Our first new term favors explanations for images that are clear and easy for the learner to understand

$$E(e_t) = \frac{1}{1 + \exp(-\beta \, \text{diff}(e_t))}. \qquad (4)$$

Here, $\text{diff}(e_t)$ represents how difficult a given explanation e_t is for image x_t, where large values indicate challenging explanations. Intuitively, the learner is more confident in discounting inconsistent hypotheses if presented with easier-to-understand explanations. This information could be crowdsourced, but later we describe a method by which both the

explanations and their interpretability can be automatically generated from image sets with ground truth labels.

Note that the quality of an explanation is not measured in the same way as image difficulty. Image difficulty in STRICT is implicitly encoded by the image's location in the feature space. Images that are close to decision boundaries are assumed to be more ambiguous to the learner compared to those that are far from decision boundaries. However, two images of equal difficulty may have very different explanations and our goal is to bias the selection of teaching images towards those that are easier to understand.

Modeling Representativeness

During teaching, we would like to select *representative* teaching images so that the concept conveyed to the learner can be easily generalized to the remaining, unseen, images. In practice however, we observe that the linear hypothesis space used by STRICT has a tendency to select outlier images (Fig. 2). When selecting the teaching set, STRICT attempts to greedily optimize the expected reduction in error for the hypothesis set and as a result can end up selecting images that may be optimal for reducing error but not necessarily informative for the learner. To overcome this limitation, inspired by approaches in active learning [29], we include an additional discount factor that favors representative examples

$$D(x_t) = \frac{1}{1 + \exp(-\gamma \operatorname{dist}(x_t))}. \quad (5)$$

Here, $\operatorname{dist}(x_t)$ encodes how dissimilar an image x_t is to other images. Again, this could be derived directly from crowdsourcing but a simple alternative is to use the density in the existing feature space

$$\operatorname{dist}(x_t) = \frac{1}{N} \sum_{n=1}^{N} ||x_t - x_n||_2^2. \quad (6)$$

In practice, we compute the mean distance for each example to all other examples in the same class. The teacher still aims to select informative examples, but this allows us to ensure that they are also representative. Setting both $\beta = \gamma = \infty$ results in the existing STRICT model.

3.4. Multiclass Teaching

Many real word teaching problems feature multiple different categories of interest. As presented, STRICT is limited to the binary class setting. One approach for handling multiple classes is to change the hypothesis space so that each individual hypothesis is a multiclass classifier *e.g.*, a softmax classifier where individual hypotheses are one-versus-all classifiers. However, generating such a multiclass hypothesis space is challenging. For instance, if one has access to a set of linear classifiers $\mathcal{H}^L$ *e.g.* crowdsourced from a

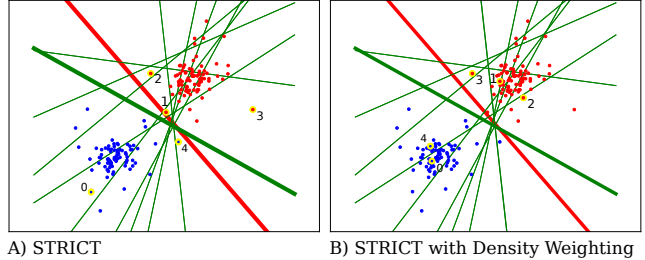A) STRICT   B) STRICT with Density Weighting

Figure 2. In A) we see that STRICT is prone to selecting outliers. Here, the numbers represent the order in which examples were selected during teaching. B) By favoring instances that are *representative* of others we select teaching examples from more dense regions of the feature space. Here, the lines represent different hypotheses, with the red one being the teachers hypothesis. The thickness of the lines represent the probability associated with the hypothesis after showing five teaching examples to the learner.

system like [36], one could assume that each multiclass hypothesis is made up of a combination of C of these linear classifiers, where C is the number of classes. This results in $|\mathcal{H}^L|!/(|\mathcal{H}^L| - C)!$ possible combinations, which becomes prohibitively large to exhaustively cover for even small numbers of hypotheses and classes.

Instead, we model a separate posterior $P_c(\mathcal{H} \mid T)$ for each of the classes. This allows us to reuse the same set of hypotheses across the different classes. It naturally enables us to model the situation where a learner can accurately group images based on their visual similarity but are unable to associate the correct class label to the groups as they have yet to learn the correct mapping. When teaching binary concepts, this corresponds to the single posterior modeling of the existing STRICT model.

3.5. Teaching Algorithm

Given our model of the learner, we now describe how we select the teaching set T. The choice of T is based on our desire to direct the learner towards a distribution over the possible hypotheses that results in the smallest number of mistakes as possible. In our multiclass setting with C classes, we use a one-versus-all scheme for each class, and maintain C different posterior distributions over the hypotheses $\mathcal{H}$. Intuitively, in order to drive down the learner's error probability in predicting multiclass labels, it is sufficient to make sure that the learner performs well in each of the C binary classification tasks. For any class c, we define the error of a one-vs-all hypothesis h over all possible images as

$$\operatorname{err}_c(h) = \frac{|x : (\hat{y}^h \neq y_c \wedge y = y_c) \vee (\hat{y}^h = y_c \wedge y \neq y_c)|}{|\mathcal{X}|}.$$

This is the fraction of images that hypothesis h disagrees with the ground truth when predicting the label y_c. After receiving the teaching set T, the expected error of the learner

for class c is defined as

$$\mathbb{E}[\text{err}_c(h) \mid T] = \sum_{h \in \mathcal{H}} P_c(h \mid T)\text{err}_c(h). \qquad (7)$$

For our multiclass teaching problem, we use the combined error probability, *i.e.*, $\frac{1}{C}\sum_c \mathbb{E}[\text{err}_c(h) \mid T]$ as a proxy for the expected error of the learner. Given a teaching budget b, we would like to find a teaching set T^*, such that upon observing the images, their labels, and associated explanations the learner would achieve the maximal reduction in expected error. Formally, let

$$R(T) = \frac{1}{C}\sum_c \left(\mathbb{E}[\text{err}_c(h)] - \mathbb{E}[\text{err}_c(h) \mid T]\right)$$
$$= \frac{1}{C}\sum_{c \in \mathcal{C}}\sum_{h \in \mathcal{H}}(P_c(h) - P_c(h \mid T))\text{err}_c(h) \qquad (8)$$

be the expected reduction in the combined error term. We aim to find

$$T^* = \underset{|T| \leq b}{\text{argmax}}\, R(T). \qquad (9)$$

Instead of directly optimizing this challenging combinatorial problem we use the greedy submodular approach outlined in [31]. We start with an empty teaching set $T = \emptyset$ and greedily add a single image at a time. The selection of the next teaching image to show amounts to choosing the example x from the unseen set ($x \notin T$),

$$x_t = \underset{x}{\text{argmax}}\, R(T \cup \{x\}). \qquad (10)$$

4. Implementation Details

In this section we outline how we automatically generate explanations, our hypothesis space, and how to efficiently optimize the teaching objective.

4.1. Image Explanations

For each image, we require an explanation e that tells the learner why an image x has a given label y. How to best generate explanations is still an open question [9]. One, time-consuming, way to acquire these explanations would be to ask an expert to manually label the informative regions in each image. Alternatively, they could be crowdsourced, but this may result in very noisy explanations *e.g.*, [8]. Instead, we propose to use the ground truth class label provided with each image to automatically generate visual explanations.

We exploit the fact that modern Convolutional Neural Networks (CNNs) used for image classification often produce semantically meaningful features. We use the Class Activation Mapping approach of [39] to automatically generate explanations for each image, but other existing methods for generating explanations can also be easily used with our

model. For each pixel location j we computed the weighted sum of the output of the final convolutional layer from a CNN that has been trained on the input data,

$$e(j) = \sum_k w_c^k f_j^k(x) + b_c. \qquad (11)$$

Here, $f_j^k(x)$ denotes the feature value at pixel location j and output channel k for the CNN f. The weight values and biases, w_c and b_c, from the final fully connected layer associated with the ground truth class are used to weight each feature map. Finally, we normalize the explanations so that each spans the range $[0, 1]$.

We use the entropy of the explanation as a proxy for the difficulty the user may have in interpreting it,

$$\text{diff}(e) = -\frac{1}{J}\sum_j e(j)\log(e(j)), \qquad (12)$$

where J is the number of pixels in the explanation. This captures our preference for localized and discrete highlighted regions whose values are either 0 or 1. In practice, weight regularization applied during the training of the CNN ensures that we do not have explanations that are uniformly high, and the classification training objective ensures that there are some non-zero entries in the feature maps to correctly predict the class label. So there is no bias towards classes that have low entropy on average, we subtract the mean difficulty for each class (making sure the final difficulties are > 0).

We use a ResNet18 [14] trained from scratch on each dataset as our backbone CNN and extract features from the output of the second residual block, resulting in a downsampling factor of 8. To this, we append an additional convolutional layer with 64 filters of size 3×3, and a final fully connected layer with the number of classes as output. We finetune the entire model on 128×128 image crops with a batch size of 64 for 60 epochs using Adam [17]. During training, we use random flips, crops, and color augmentations. We start with an initial learning rate of 0.0002 and decay by a factor of 10 after 30 epochs. To generate the final explanation map e for each 144×144 input image x, we extract the explanation for the center 128×128 crop and pad and resize it to input resolution.

4.2. Hypotheses Space Generation

We require access to a representative hypothesis space $\mathcal{H}$ to perform teaching. This space should span the different possible linear classifiers that the learners may be using. It represents the different biases that they may have in relation to the teaching task at hand. However, generating such a space for multiple fine-grained categories is not trivial. As an alternative to generating embeddings from crowd annotations *e.g.* [36], we propose to use the features from our finetuned CNN from the previous section. Extracting the

64 dimensional penultimate feature vector from our modified ResNet18 gives us a representation that encodes visual similarity. While this is unlikely to be the same representation used by our learners it has the advantage that it can be generated with no additional annotation cost.

We generate a set of candidate linear classifiers using the CNN features by first clustering the features in each class into 2 subsets and training a linear SVM to separate each from the rest of the data. We also train a linear SVM to separate each class from the other classes, representing the best possible hypothesis h^*. Next, we group each pair of classes and train additional linear models to separate these from the remaining classes. We add all these linear classifiers to the hypothesis set and add an additional random linear classifier to bring the total number of hypotheses to 100. Each dataset is split into 80% training and 20% test sets. As in [31], images in the training set that are not possible to correctly classify with the optimal hypothesis h^* are removed.

4.3. Efficient Optimization

Optimizing Eq. (10) involves searching over all unseen images in $\mathcal{X}$ and measuring their reduction in error for each hypothesis in $\mathcal{H}$ across all classes. With the aid of some pre-computation, we can reduce the updates to simple matrix operations. For simplicity, we present this in terms of the binary STRICT model [31], but the same formulation holds for our multiclass EXPLAIN algorithm by including the additional discount factors and classes.

First, we note that as the prior term in Eq. (8) is constant at each time step and can be removed without affecting the selection of x_t. Second, we can re-factor the posterior into the current posterior multiplied by the update

$$R(T \cup \{x\}) = -\sum_{h \in \mathcal{H}} \text{err}(h)P(h \mid T)\delta(h, x), \quad (13)$$

where $\delta(h, x) = P(y|h, x)$ if $y \neq \hat{y}^h$ and otherwise it is set to 1. In matrix notation, this can be rewritten as

$$r = -(e \circ p_i)L, \quad (14)$$

where $\circ$ is the element-wise multiplication. e is a vector of size $1 \times |\mathcal{H}|$ that contains the error associated with each hypothesis, and can be computed once offline. p_i is a vector of size $1 \times |\mathcal{H}|$ that represents the current posterior over the hypotheses, which we will update after each time step i. L is a $|\mathcal{H}| \times |\mathcal{X}|$ matrix that can be computed once offline. Each entry in L encodes the confidence each h places in each example x

$$L_{h,x} = \begin{cases} 1, & \text{if } y = \hat{y}^h \\ P(y \mid h, x), & \text{otherwise.} \end{cases} \quad (15)$$

Computing the next teaching example, x_t simply amounts to choosing the maximum entry in the $1 \times |\mathcal{X}|$ error reduction vector r, taking care to only select from the examples not currently in the teaching set. We then update the posterior using

$$p_{i+1} = p_i \circ l_t, \quad (16)$$

where $l_t = L_{:,t}$ is the column vector from L associated with the selected teaching example x_t. For the multiclass setting, we simply maintain a separate e^c, L^c, and p^c for each class and choose the next teaching example that results in the biggest reduction across all classes.

5. Experiments

5.1. Datasets

There are existing datasets with explanations in the form of annotated visual attributes that could be used to teach visual categories to human learners. However, we found that in many cases, $e.g.$, [35], the provided attributes were too coarse-grained ($e.g.$ 'forehead color') to be useful for teaching highly similar categories. Other alternatives were too noisy to be informative $e.g.$ [8], but present an interesting avenue for future human-in-the-loop explanation generation. Instead, we selected three diverse datasets to span a range of different teaching use cases. Example images and explanations from each dataset can be seen in Fig. 3.

Butterflies Our first dataset represents the teaching of naturalists who wish to identify different species of plants and animals in the wild. It contains images of five different species of butterflies captured in a large variety of real-world situations with varying image quality from the iNaturalist dataset [33]. Two of the species, Monarch and Viceroy, are very similar in appearance and the third, Queen, looks the same from underneath. The final two species, the Red Admiral and the Cabbage White are distinct in appearance. We extracted a bounding box around each butterfly and manually discarded images of caterpillars. In total, the dataset contains 2,224 images, close to uniformly distributed across each of the species.

OCT Eyes Our second dataset mimics the teaching of a trainee ophthalmologist that is tasked with learning how to identify different retinal diseases in images. It consists of image slices of the retina obtained through optical coherence tomography (OCT). OCT is a non-invasive imaging technique that uses light waves to take cross-sectional images of the retina. It is commonly used in ophthalmology, where on the order of 30 million OCT scans are performed around the world annually [32]. Here, the learner's goal is to classify each of the images into one of three classes: Normal, contains Macular Edema, or contains Subretinal Fluid. Diabetic Macular Edema is one of the leading causes of vision loss for people with diabetes [28]. Many of the examples are challenging and require subtle inspection to correctly identify the condition. The dataset contains 1,125 images with ground truth annotations provided by retina specialists.
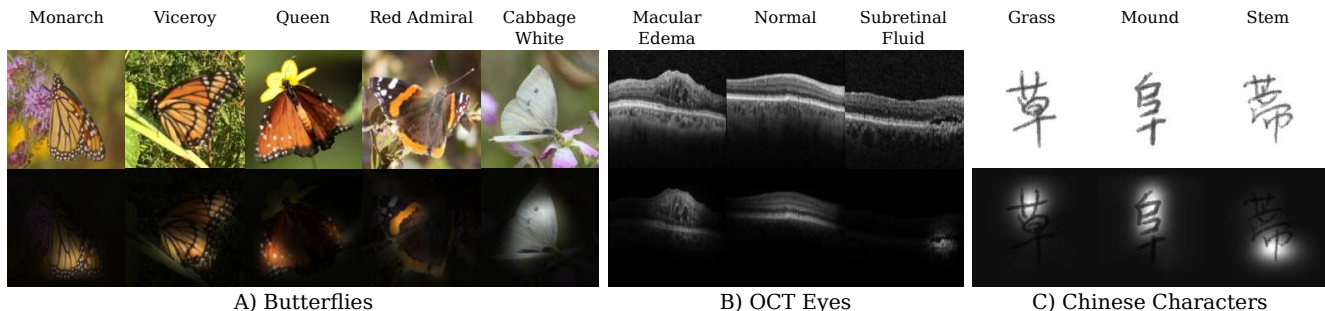
| Monarch | Viceroy | Queen | Red Admiral | Cabbage White | Macular Edema | Normal | Subretinal Fluid | Grass | Mound | Stem |

A) Butterflies B) OCT Eyes C) Chinese Characters

Figure 3. Here we see example images from our three datasets with their corresponding, automatically generated, visual explanations below.

Chinese Characters Finally, we use a dataset of three different Chinese characters extracted from [20], that was also used in [16]. Here, we are exploring the scenario of a non-native Chinese speaker attempting to learn to identify new characters. Each of the 717 images is a different handwritten example of one of the following three characters: Grass, Mound, or Stem. The images vary in difficulty as there is a large variety in the handwriting quality and style for the different individuals that contributed to the dataset.

5.2. Experimental Setup

We conducted experiments with real human participants on the crowdsourcing platform Mechanical Turk using the same methodology as [31, 16]. Crowd workers, *i.e.* learners, were randomly assigned to a dataset and baseline teaching strategy. On average we received 40 participants per strategy and dataset combination. We assumed that the learners were motivated to perform well but acknowledge that this might not always be the case. First, the learners were shown a brief tutorial illustrating how our web-based teaching interface worked. When teaching began, they were presented with a sequence of images, shown one at a time, selected by the assigned teaching algorithm. After viewing each image in turn, they were then asked to select from a list of options indicating the class they believed the image to contain. Depending on the assigned teaching strategy, they were given feedback either in the form of the correct class label, or the correct class label and its corresponding visual explanation. The visual explanations were displayed on top of the input images, alternating between the input image and the explanation every 0.5 seconds. Upon receiving feedback, learners had to wait for a minimum of 2 seconds before they could proceed to the next teaching image. For each dataset we showed 20 teaching images, followed by 20 randomly selected testing images where no feedback was provided during testing. This random sequence of test images was different for each learner. Longer teaching sequences would likely lead to better test performance but there is a trade off between the learner's attention span and their accuracy. We shuffled the order of the response buttons depicting the class labels for each learner before teaching began to ensure that there was no bias towards a particular button position.

	RAND_IM	RAND_EXP	STRICT	EXPLAIN
Butterflies	65.20	67.31	65.00	**68.33**
OCT Eyes	51.05	62.58	64.63	**72.38**
Chinese	47.05	**58.90**	51.91	46.35
Chinese - Crowd			53.06	**65.44**

Table 1. Average test time accuracies for Mechanical Turk learners across the three different test datasets. Compared to the standard image only baseline RAND_IM, the inclusion of our explanations in RAND_EXP results in better test time performance across all datasets. Our EXPLAIN model results in superior learners in two of the three datasets with CNN generated hypothesis spaces.

We compared our full EXPLAIN model to three baseline algorithms: 1) RAND_IM random selection of images, 2) RAND_EXP random selection with visual explanations, and 3) our multiclass version of STRICT [31]. Note, this last baseline did not include density weighting or explanations. For both EXPLAIN and STRICT we generated a fixed teaching sequence once offline and used it for all learners with the same hypothesis space for both. We set the noise level of the learners to $\alpha = 0.5$, and the explanation and representativeness parameters to $\beta = \gamma = 1.0$.

5.3. Results

In Table 1 we report the average test time accuracy for each learner for the different baseline teaching strategies and datasets. Fig. 4 A) - C) displays the histograms of these accuracies illustrating that strategies that include explanations tend to do better overall.

For the Butterflies dataset we see that a greater percentage of learners get high scores when taught with EXPLAIN, see Fig. 4 A). This is a challenging dataset, and we observe that many learners remain confused between the three similar species by the end of teaching, perhaps necessitating longer teaching sequences.

The OCT Eyes images are likely to be the most unfamiliar to our learners compared to the other datasets. However, the images are well aligned and do not contain large out of plane view point changes or confusing background texture. This enables us to generate high-quality explanations that localize the characteristic morphologies for the different retinal diseases. For the RAND_IM baseline in Fig. 4 B) the difference in learner performance can be explained by the images selected and their intrinsic motivation. In Fig. 4 D)
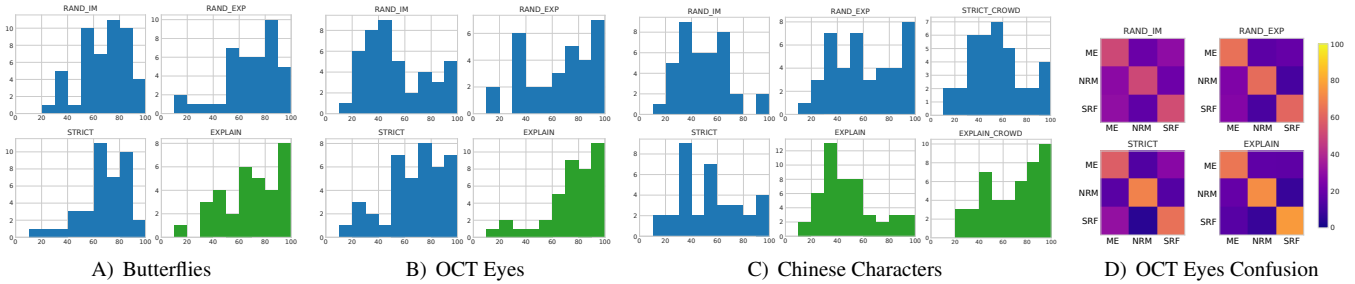
Figure 4. Test time classification performance for human learners. A) - C) Show learners binned by test time accuracy. The horizontal axes represent average test scores, where larger numbers indicate higher accuracy. The vertical axes are the number of learners. D) Test time learner confusion matrices for the OCT Eyes dataset. We see that both explanation based strategies result in smaller off-diagonal entries.

we see the average test time confusion matrices across all learners for the OCT Eye dataset. It is clear that EXPLAIN results in less cross-category confusion *i.e.* smaller values in the off-diagonal entries. We see that learners tend to make fewer mistakes when identifying Subretinal Fluid as it is relatively distinct, while there is more confusion between Macular Edema and Normal.

The Chinese Characters dataset represents an interesting failure case for EXPLAIN when using the CNN generated hypothesis space, see Fig. 4 C). In Fig. 5 we see the average performance during teaching for all learners and the first five teaching images selected by EXPLAIN. EXPLAIN selects a particularly difficult image for the fourth teaching example. It happens to be the same class as the previous image, but is visually very different. From inspecting the test time confusion matrix we see that this early difficult example potentially biases learners as they end up with lower performance for the *Grass* class. The random based strategies have the advantage of generating different teaching sequences for each learner by uniformly sampling the input space thus introducing some additional robustness. In Table 1 we see that RAND_EXP performs well on this dataset, indicating that the performance dip for EXPLAIN may be a result of the automatically generated hypothesis space or the explanation interpretability scores, rather than the quality of the explanations themselves. To test this hypothesis, we assigned manual interpretability scores and generated a separate embedding space more closely aligned with human notions of similarity by soliciting pairwise similarity estimates on Mechanical Turk to construct a new embedding space (Chinese - Crowd)[1]. Teaching with this embedding space (EXPLAIN_CROWD) results in the best test time performance overall, see the last row in Table 1.

6. Conclusion

We introduced EXPLAIN, an algorithm for teaching multiple visual categories to human learners with interpretable feedback. Interpretable feedback can be applied to any exist-
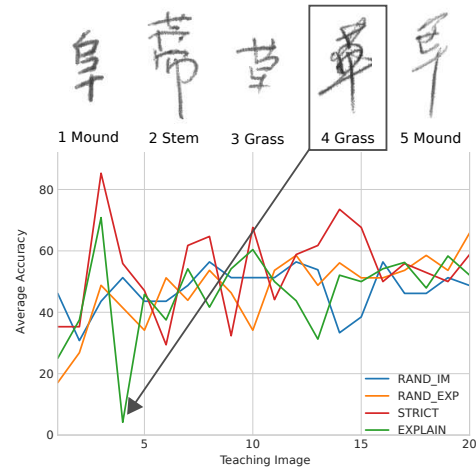

Figure 5. Average accuracy during teaching for the Chinese Characters dataset using the CNN generated hypothesis space, where random guessing is 33%. For all strategies, we observe a general improvement in learners' ability over time. Above we see the first five teaching images selected by EXPLAIN. The fourth image is a difficult instance, resulting in the vast majority of learners guessing the wrong category.

ing visual teaching algorithm. It provides cues to the learner during teaching in the form of visual explanations and explicitly models how they incorporate this information when updating their belief. This enables us to generate more informative teaching sequences resulting in improved test time performance *i.e.*, better generalization from the learners on unseen images. Experiments featuring real human participants show that our approach is superior to existing methods that only provide weaker class label feedback. In future, we plan to investigate extending our approach to the interactive teaching setting, where the model updates online based on the learner's responses *e.g.*, [30, 10, 16].

---

[1]Full details are provided in the supplementary material.

References

[1] J. H. Bak, J. Y. Choi, A. Akrami, I. Witten, and J. Pillow. Adaptive optimal training of animal behavior. In *NIPS*, 2016. 2

[2] S. Basu and J. Christensen. Teaching classification boundaries to humans. In *AAAI*, 2013. 2

[3] T. Berg and P. N. Belhumeur. How do you tell a blackbird from a crow? In *ICCV*, 2013. 2

[4] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Welinder, P. Perona, and S. Belongie. Visual recognition with humans in the loop. *ECCV*, 2010. 2

[5] W. Canfield. Aleks: A web-based intelligent tutoring system. *Mathematics and Computer Education*, 2001. 1

[6] Y. Chen, O. Mac Aodha, S. Su, P. Perona, and Y. Yue. Near-optimal machine teaching via explanatory teaching sets. In *AISTATS*, 2018. 2

[7] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 1994. 2

[8] J. Deng, J. Krause, M. Stark, and L. Fei-Fei. Leveraging the wisdom of the crowd for fine-grained recognition. *PAMI*, 2016. 2, 5, 6

[9] B. Doshi-Velez, Finale; Kim. Towards a rigorous science of interpretable machine learning. In *arXiv*, 2017. 5

[10] J. Du and C. X. Ling. Active teaching for inductive learners. In *SIAM International Conference on Data Mining*, 2011. 8

[11] B. S. Eaves Jr, A. M. Schweinhart, and P. Shafto. Tractable bayesian teaching. 2015. 2

[12] S. Goldman and M. Kearns. On the complexity of teaching. *Journal of Computer and System Sciences*, 1995. 2, 3

[13] E. R. Grant and M. J. Spivey. Eye movements and problem solving: Guiding attention guides thought. *Psychological Science*, 2003. 2

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[15] L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, and T. Darrell. Generating visual explanations. In *ECCV*, 2016. 2

[16] E. Johns, O. Mac Aodha, and G. J. Brostow. Becoming the expert-interactive multi-class machine teaching. In *CVPR*, 2015. 1, 2, 7, 8

[17] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv*, 2014. 5

[18] K. R. Koedinger, J. R. Anderson, W. H. Hadley, and M. A. Mark. Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education*, 1997. 1

[19] H. Lakkaraju, S. H. Bach, and J. Leskovec. Interpretable decision sets: A joint framework for description and prediction. In *KDD*, 2016. 2

[20] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang. Casia online and offline chinese handwriting databases. In *International Conference on Document Analysis and Recognition*, 2011. 7

[21] W. Liu, B. Dai, J. M. Rehg, and L. Song. Iterative machine teaching. *ICML*, 2017. 2

[22] R. Peterson. A field guide to the birds: a completely new guide to all the birds of eastern and central North America, 1998. 2

[23] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. In *NIPS*, 2015. 2

[24] S. Poulis and S. Dasgupta. Learning with Feature Feedback: from Theory to Practice. In *AISTATS*, 2017. 2

[25] A. N. Rafferty, E. Brunskill, T. L. Griffiths, and P. Shafto. Faster teaching by pomdp planning. In *AIED*, 2011. 2

[26] M. T. Ribeiro, S. Singh, and C. Guestrin. Why should I trust you?: Explaining the predictions of any classifier. In *KDD*, 2016. 2

[27] B. Roads, M. C. Mozer, and T. A. Busey. Using highlighting to train attentional expertise. *PloS one*, 2016. 2

[28] P. Romero-Aroca. Managing diabetic macular edema: The leading cause of diabetes blindness. *World journal of diabetes*, 2011. 6

[29] B. Settles. Active learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 2012. 4

[30] A. Singla, I. Bogunovic, G. Bartók, A. Karbasi, and A. Krause. On actively teaching the crowd to classify. In *NIPS Workshop on Data Driven Education*, 2013. 8

[31] A. Singla, I. Bogunovic, G. Bartók, A. Karbasi, and A. Krause. Near-optimally teaching the crowd to classify. In *ICML*, 2014. 1, 2, 3, 5, 6, 7

[32] E. Swanson and D. Huang. Ophthalmic OCT reaches $1 billion per year. *Retinal Physician*, 2011. 6

[33] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie. The iNaturalist Species Classification and Detection Dataset. In *CVPR*, 2018. 6

[34] L. von Ahn. Duolingo: learn a language for free while helping to translate the web. In *International Conference on Intelligent User Interfaces*, 2013. 1

[35] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 6

[36] P. Welinder, S. Branson, P. Perona, and S. J. Belongie. The multidimensional wisdom of crowds. In *NIPS*, 2010. 4, 5

[37] J. Whitehill and J. Movellan. Approximately optimal teaching of approximately optimal learners. *Transactions on Learning Technologies*, 2017. 2

[38] Q. Zhang, Y. N. Wu, and S.-C. Zhu. Interpretable convolutional neural networks. *arXiv*, 2017. 2

[39] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 2, 5

[40] X. Zhu. Machine teaching for bayesian learners in the exponential family. In *NIPS*, 2013. 2