Explainable Interactive Projections for Image Data

Huimin Han[1], Rebecca Faust[1], Brian Felipe Keith Norambuena[1], Ritvik Prabhu[1], Timothy Smith[1], Song Li[1], and Chris North[1]

Virginia Tech, Blacksburg VA 24061
{huimin,rfaust,briankeithn,ritvikp,tim23,songli,north}@vt.edu.

Abstract. Making sense of large collections of images is difficult. Dimension reductions (DR) assist by organizing images in a 2D space based on similarities, but provide little support for explaining why images were placed together or apart in the 2D space. Additionally, they do not provide support for modifying and updating the 2D space to explore new relationships and organizations of images. To address these problems, we present an interactive DR method for images that uses visual features extracted by a deep neural network to project the images into 2D space and provides visual explanations of image features that contributed to the 2D location. In addition, it allows people to directly manipulate the 2D projection space to define alternative relationships and explore subsequent projections of the images. With an iterative cycle of semantic interaction and explainable-AI feedback, people can explore complex visual relationships in image data. Our approach to human-AI interaction integrates visual knowledge from both human mental models and pre-trained deep neural models to explore image data. We demonstrate our method through examples with collaborators in agricultural science.

Keywords: Interactive Dimension Reduction, Semantic Interaction, Explainable AI, Image Data

1 Introduction

People commonly use dimension reduction (DR) methods to explore data for sensemaking tasks [8]. DR methods excel at mapping high-dimensional data to a low-dimensional space (typically 2D) while preserving meaningful structure and relationships. Several methods add interaction to enable exploration, modification and understanding of the 2D space. For example, some systems incorporate semantic interactions which couple cognitive and computational processes by inferring meaning behind interactions and updating the model accordingly [12].

However, most of interactive DR methods have limited support for image data, often representing images as arrays of pixels and treating them the same as tabular data. This not only limits the DR's ability to determine similarities between images, but also often inhibits interaction methods for understanding the 2D space. For example, Self et al.'s Andromeda uses Weighted Multidimensional Scaling (WMDS) to create an interactive DR that supports semantic interaction for exploring and understanding 2D projection spaces via model steering [29]. After an interaction, the model learns new weights on the input dimensions that infer meaning from the interaction and explain

the information learned by the projection. However, when a dataset does not have interpretable dimensions, these explanations become meaningless. What's more, because a single pixel has an arbitrary meaning across all images, weighting the same pixel in each image does not have a uniform effect on all of the images. Thus it does not make sense to directly project images from pixel arrays.

We know from past research that deep neural networks excel at extracting meaningful features from images and embedding them into a new representation [7]. Classifiers commonly use these embeddings, achieving high accuracy which indicates that the embeddings must be well suited for finding similarities between images. The question then remains, how can we use these feature embeddings to create more meaningful projections of image data and capture human feedback?

In this paper, we present an interactive DR method, built from Self et al.'s Andromeda, that supports semantic interaction for exploring projections of image data. Our method leverages the feature embeddings extracted from a convolutional neural network to project image data to a low-dimensional space using WMDS, while supporting semantic interaction to enable people to explore and update the projection space. Our method enables people to directly manipulate the 2D locations of images to define new pairwise relationships in the 2D space and then learns new projection weights that best respect those relationships. Using these weights to re-project the images, people can observe impact of those relationships on the projection space. Each dimension now represents some feature of the images, rather than an arbitrary pixel, but are still not directly interpretable. Increasing the weight on a feature increases its importance in the projection but still does not provide any insight into the information learned. Thus, while updating the weights now has inherent meaning, people have no real understanding of this meaning. That brings us to our second question: how can we translate the learned weights back to the image space?

In addition to providing an interactive DR, our approach provides explanations of features of importance in the 2D space through the use of a weighted backpropagation algorithm. We adapt a traditional visual backpropagation method for generating saliency maps [4] to apply the feature weights from the projection. Doing so creates saliency maps that emphasize the image features most influential to the projection's placement of the image. Thus, we are able to push the information learned from the human interaction back through the network to the image space, where people can interpret it.

Our method helps people explore multiple projections of their image data through semantic interactions and explain the effects of these interactions on the placement of images through saliency maps. Figure 1 presents an example using our method.

The contributions of this paper include:

- An interactive-AI method for dimension reduction that semi-automatically projects images based on visual knowledge from both pre-trained neural models and human feedback.
- An explainable-AI method for saliency mapping through weighted backpropagation that explains important image features.
- A usage scenario, built from our collaboration with agriculture sciences, illustrating a real world example of image exploration tasks supported by our methods.

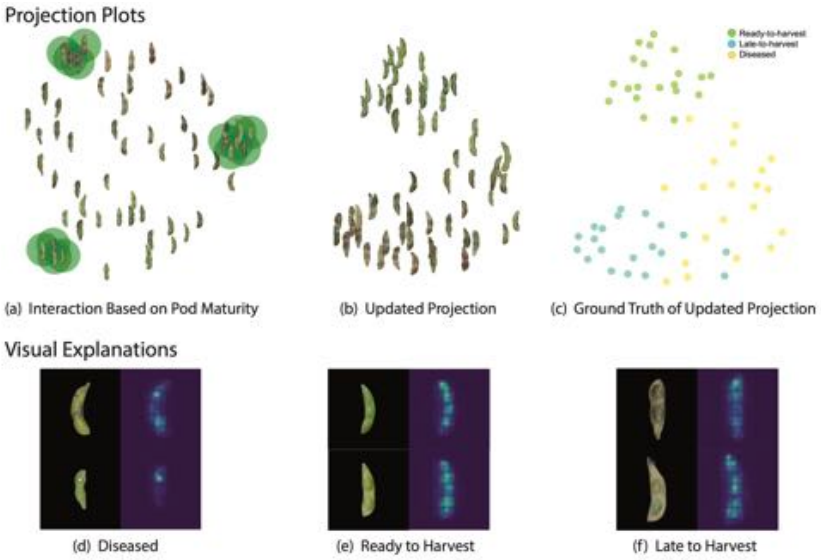

Fig. 1. Interactions to explore maturity level in edamame pod images. (a), shows user manipulations based on maturity level. (b) shows the updated projection while (c) shows the ground truth maturity level. (d) - (f) shows the explanations of important image features for each maturity level.

2 Related Work

Our work draws elements from interactive dimensionality reduction techniques, semantic interaction methods, and explainability in deep learning. In this section, we start by discussing related works from the interactive dimensionality reduction literature. Next, we focus on semantic interaction and its applications in sensemaking. Finally, we discuss explainability techniques for deep learning methods in the context of image data.

2.1 Interactive Dimensionality Reduction

Dimensionality reduction techniques are commonly employed to analyze and visualize high-dimensional data by projecting it onto a 2D or 3D space [31]. Alone, DR algorithms typically produce a static projection space with no means for exploration or manipulation. Thus, many scholars sought to develop *interactive* DR techniques capable of capturing user feedback and subsequently modifying the projection.

Some interactive DR methods create a bi-directional workflow where people can alter data in the high dimensional space to see the effect on the 2D location and vice versa [6,22]. Other works explore the idea of backwards (or inverse) projections that allow people to select locations in the 2D space and generate corresponding high-dimensional representations [16,28]. PEx-Image specifically targets image data, providing interactions for exploratory tasks, such as zooming into specific projection regions, displacing points to resolve overlapping and displaying nearest neighbors of selected images [11].

Many works exist on interactively steering projections. Several take the approach of requiring people to define control and organize control points, which are then used to project a larger collection of data while maintaining local structures around control points [23, 25, 26]. Others learn new distance functions for MDS to update the projection to best respect user manipulations [5, 29]. Fujiwara et al. provide a visual analytics framework for comparative analysis, providing interactions to manipulate and update projections to illustrate the similarities and differences between clusters of points [17].

Our work expands on past work by specifically targeting imaged data to provide both projection-steering interactions and visual explanations of the 2D space. We extend Self et al.'s Andromeda [29]. Andromeda allows people to directly manipulate the 2D location of data points and updates the projection model to incorporate human feedback into the projection. We propose an extension to Andromeda that supports image data via deep learning feature representations and provides visual explanations of the important image features, before and after human feedback.

2.2 Semantic Interaction

Semantic interactions exploit the natural interactions in visualizations to learn the intent of the user and then, based on these interactions, update the underlying model and its parameters [14]. In the context of sensemaking, semantic interactions capture the analytical reasoning of the users [13], and support analysts throughout the sensemaking process [10].

Most semantic interaction systems work using a dimensionality reduction model, similar to the interactive dimensionality reduction methods described in the previous section. Semantic interaction is a bidirectional pipeline [9] and requires capturing the changes in the visualization and turning them into changes to the model. In the dimensionality reduction case, this is usually done through the use of an inverse transformation (e.g., inverse WMDS) [33]. There are several models that can be used to solve the bi-directional transforms required to implement semantic interactions, such as Observation-Level Interaction [15], Bayesian Visual Analytics [21], and Visual to Parametric Interaction [24].

Previous work has also shown how to integrate deep learning models with semantic interaction techniques. Bian and North [1] developed a semantic interaction model for text analytics integrating traditional dimensionality reduction techniques with a BERT neural network as its core component. Bian et al. [2] continued the development of these semantic interaction models and designed an explainable AI framework based on counterfactuals that help users understand the generated projection.

2.3 Explainability in Deep Learning

Scholars have proposed several explainability methods for convolutional neural network (CNN) models, the backbone of most image-based deep learning applications. Bojarski et al. [4] proposed a visualization method that shows which pixels of an input image contribute the most towards the predictions of a CNN model. In particular, their technique allows debugging CNN-based systems by highlighting the regions of the input image that have the highest influence on the output of the model. Zeiler and Fergus [35] developed a novel visualization technique that provides insight into the intermediate feature layers

of a CNN in a classification task. Zhou et al. [36] use a global average pooling layer to shed light on how this layer enables CNN models to localize objects in images. In particular, their approach generates a Class Activation Map (CAM) using global pooling. However, while these explanation techniques are powerful, they are designed for specific CNN-based models. To address this weakness, researchers have proposed visual explanation techniques for a large class of CNN-based models. For example, Selvaraju et al. [30] generated CAMs based on gradient information of target concepts (Grad-CAM). Grad-CAM provides fine-grained explanations of the CNN predictions, but suffers from performance issues with multiple occurrences and single-object images.

Despite the recent advances in explainable deep learning for image data, there is a dearth of studies exploiting explainable deep learning techniques for interactive DR in the context of image analysis. Thus, our work seeks to fill this gap and combine interactive DR for images with explainable deep learning techniques. In particular, we base our work on the method of Bojarski et al. [4], as visual backpropagation provides an efficient way to generate explanations of relevant image features for the users by pushing the weights obtained in the interactive DR loop through the backpropagation process.

3 Tasks

Before discussing the details of our method, we first must discuss the sensemaking tasks of someone using our it it ->method or tool?. Pirolli and Card described the sensemaking process as having two primary loops: the foraging loop and the sensemaking loop [27]. The foraging loop focuses on searching and filtering information and extracting evidence. The sensemaking loop then uses this information to iteratively construct representational schemas as well as generate and test hypotheses about the data.

In the context of image data, simply looking at every image does not provide sufficient information to make sense of the data. The foraging loop requires filtering and extracting sets of images relevant to the task at hand. Then, those images must be organized into a schema that provides a structured representation for consuming the image data and testing hypotheses. The process of generating and refining the schema typically requires several iterations of foraging for information under the current schema, updating the schema based on the new information, and evaluating how the schema fits the task at hand to determine if it requires further refinement.

Our method supports this schematization step through iterative exploration of the images and refinement of the 2D representation to reflect prior knowledge of the analysis task. Through discussions with collaborators in the plant sciences, we identified the following tasks to support this iterative process: (1) Define custom similarities based on prior knowledge and (2) link human and machine defined similarities

These tasks create a synergy between the machine and the human where they work together as a team, teaching each other what they have independently learned from the data. In the end, we create an analysis pipeline where the human perceives the data, conveys their knowledge to the machine, and the machine then re-organizes the data based on this information, while providing explanations of its reasoning. The remainder of this section discusses these tasks in greater detail.

3.1 Define custom similarities based on prior knowledge

When analyzing data, people typically have some prior knowledge about the data, such as what categories of or similarities between images they expect to exist within the data. For example, in a set of edamame pod images, the analyst may expect images of healthy pods and diseased pods. Static dimension reduction plots, may or may not adequately reflect this prior knowledge. In the previous example, the person analyzing may want to inspect healthy vs diseased pods, but the model may not naturally recognize these differences. Furthermore, static projections do not enable people to explore different projections defined under different guidelines. To enable hypothesis testing, people must be able to steer the projection to define similarities in the data in a way that reflects their prior knowledge. With our method, people directly manipulate the 2D location of images to define new relationships within the data that the model then learns and uses to re-project the images accordingly.

3.2 Link human and machine defined similarities

The previous task focuses on teaching the projection model to incorporate human knowledge. However, while it helps the model learn human knowledge, it does not help people understand the model's knowledge. People need ways to inspect the image features most important to the 2D projection. This helps them not only understand the 2D space, but also validate the models perception of their knowledge and potentially identify other image similarities/differences beyond the knowledge they taught the model. Our method provides saliency maps that illustrate the features of the image that the projection most heavily used to place the image. Viewing the explanations of multiple images provides insight into why the model placed them near or far from each other and provides a means for understanding the 2D space.

4 Workflow and Methodology

In this section, we describe the expected user workflow and interactions, as well as the underlying methodology. Figure 2 gives an overview of the workflow while Figure 1 presents an example of using this workflow.

4.1 Initial State

Upon loading the data, our method extracts image features to project. It then uses Weighted Multidimensional Scaling (WMDS) to project the features into 2D which provides the initial view of the data and a starting point for the exploratory analysis. We chose WMDS because it uses pairwise similarities as the input for projection and thus changes in the 2D similarities conceptually map directly back to the input space.

Feature Extraction Recently, deep learning models have become popular for feature extraction in images [18]. In particular, Convolutional Neural Networks (CNN) have shown great power in image-related tasks and as a result using CNNs has become the

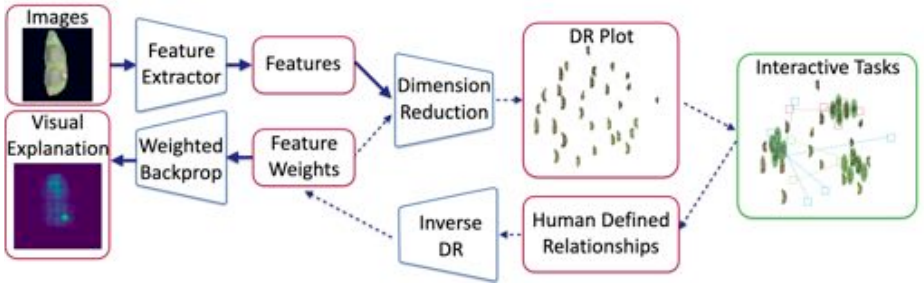

Fig. 2. An overview of our workflow. First, we extract image features using a deep learning feature extractor which we then pass to an interactive DR method (WMDS) that facilitates semantic interactions. After interactions, we pass the newly defined relationships to the inverse DR where it learns new projection parameters that best respect them and re-projects the images.

standard in feature extraction [32]. For our research, we use pre-trained ResNet18 [20] as a fixed feature extractor to generate features vectors from images.

Given an image dataset $\mathscr{D}$, we forward propagate the images through the network with the fully connected layer removed. The final representations are denoted as $\mathscr{X} = ResNet_{\mathsf{pre-trained}}(\mathscr{D})$. The feature space $\mathscr{X}$ is a 512-dimensional space used to represent the images. Each image representation ($x_i \in \mathscr{X}$) is the output of applying average pooling to the final feature map of the network. We use $\mathscr{X}$ as the input to the interactive dimension reduction loop.

Weighted Multidimensional Scaling Using the features extracted from the images ($\mathscr{X}$) as input, we perform MDS on a weighted data space to project the images to 2D, using the following function:

$$y = \underset{y_1,\ldots y_n}{\arg\min} \sqrt{\sum_{i<j\leq N} (dist_L(y_i,y_j) - dist_H(w,x_i,x_j))^2} \tag{1}$$

where N is the number of points in the dataset, $dist_L(y_i,y_j)$ is the low-dimensional distance between y_i and y_j and $dist_H(w,x_i,x_j)$ is the weighted high dimensional distance between the feature representations x_i and x_j, given the dimension weights w.

For the initial projection, we initialize w with equal weights for every dimension, relying solely on the raw image features to organize the images.

4.2 Interactions & Inverse Projection

After the initial projection, our method allows people to directly manipulate the projection plot, dragging points into new positions in the 2D space. Manipulated points define new pairwise relationships for the projection model to learn during the inverse projection. Once the analyst completes their interaction, the model uses these relationships to optimize the projection weights to create a layout that best respects the defined relationships.

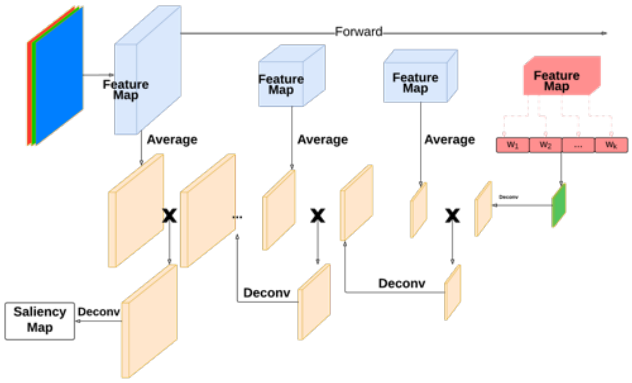

Fig. 3. Weighted Visual Backpropagation Process

Interactive Dimension Reduction To facilitate interactive dimension reduction, we use inverse WMDS (WMDS^{-1}) to update the projection after semantic interactions, as originally described in Andromeda [29].

After a person re-positions a subset of the points, y^*, we perform WMDS^{-1} to calculate new weights optimal for maintaining the specified relationships, thus capturing human feedback. WMDS^{-1} uses the following equation to update the weights:

$$w = \underset{w_1,\ldots w_d}{\arg\min} \sqrt{\frac{(\sum_{i<j\leq N}(dist_L(y_i^*, y_j^*) - dist_H(w, x_i, x_j))^2}{\sum_{i<j\leq N} dist_H(w, x_i, x_j)}} \tag{2}$$

This equation produces a vector of dimension weights that best respects the 2D pairwise similarities specified through the interactions. We normalize the weight vector to sum to 1, so as to normalize the HD distances to a roughly constant sized space. We then re-project the images using equation 1 with the updated weights to create a layout that incorporates human feedback.

4.3 Visual Explanations

Our method also provides visual explanations in the form of saliency maps that highlight the important features for projecting a given image, shown in Figure 1(d)-(f). In these maps, the brighter pixels correspond to features of greater importance.

In the initial view, before semantic interactions, these explanations indicate the features of importance identified by the feature extractor that the projection model then uses to place the images. After an interaction, the optimized parameters are pushed backwards through the feature extractor, using weighted backpropagation, to generate new saliency maps that emphasize the features learned by the projection model. By inspecting the differences between the original saliency map and the post-interaction map, people can understand what features the projection learned from their interaction.

Weighted Visual Backpropagation Figure 3 illustrates our weighted visual backpropagation method. We base our proposed method on the visual backpropagation method proposed by Bojarski et al. [4]. This method computes the actual contribution of neurons to the feature representation, making the backpropagation fast and efficient. We make this method projection-aware by applying the projection weights to the backpropagation.

To implement our method, we utilize the feature maps after each ReLU layer. For the feature map of the last convolutional layer, we conduct channel-wise multiplication with the weights w obtained from the interactive DR loop to back-propagate the user's intent. We then average the other feature maps to get a single feature map per layer. The deepest single feature map, highlighted in green in Figure 3, is deconvolved with the same filter size and stride as the convolutional layer immediately preceding it. This scales the feature map to match the size of the map in the previous layer. Then we point-wise multiply the deconvolved feature map by the averaged single feature map of the previous layer. This process is repeated until we reach the input image.

We keep our notation consistent with Bojarski et al. [4]. Note, we will only describe our modification to their method. For full details, please refer to Bojarski et al. Consider a convolutional neural network $\mathscr{N}$ with n convolutional layers. Let $\gamma(i)$ denote the value of pixel i of the input image and v represent a neuron. e represents an edge from some other neuron v' to v and a_e denotes the activation of v ($a_e = a(v)$). $\mathscr{P}$ denotes a family of paths. The contribution of the input pixel i, calculated by the original Visual Backpropagation method, is defined as:

$$\theta^{\mathscr{N}}_{VBP}(i) = c * \gamma(i) \sum_{P \in \mathscr{P}} \prod_{e \in P} a_e \tag{3}$$

To back-propagate the weighted feature map, we conduct channel-wise multiplication for the last feature map with weights gained from the interactive DR loop. We denote et as the edge that connects nodes from the layer $(t-1)$ to the layer t. Let k denote the kernels for each layer. The contribution of the input pixel i calculated by our Weighted Visual Backpropagation method is defined as

$$\theta^{\mathscr{N}}_{WVBP}(i) = c * \gamma(i) \sum_{P \in \mathscr{P}} \prod_{e \in P} a_{et} \tag{4}$$

where

$$a_{et} = \begin{cases} a(v) & \text{if t}\neq\text{n}, \\ a(v) * w_k & \text{if t=n}. \end{cases}$$

and w_k is the weight from the inverse projection corresponding to channel k of the feature map in the final layer.

5 Usage Scenario: Edamame Pods

We developed this usage scenario with our collaborators in the plant sciences department [19]. Our collaborators identified the need for incorporating human perception into model development for identifying plant features. Initially, they wanted to organize images of edamame pods based on maturity level. However, when sorting the images

they also discovered that the pods contained varying numbers of seeds, which often correlates to the consumers' perception of quality. They envisioned that a method like ours would help them re-organize the images based on this newly identified feature and allow them to reuse the original model. In the remainder of this section, we discuss two scenarios for organizing images of edamame pods. For our example, we use a subset of their edamame pod dataset containing 60 images, with 20 images per maturity stage.

Maturity Stage The maturity stage of each pod is defined as either diseased, late-to-harvest, or ready-to-harvest. Here, we test if our method can sort the images according to these phenotypes and whether the features captured by the model to separate the images are related to the underlying phenotypes, illustrated in Figure 1. First, we project the edamame pods to 2D. Then, we observe the visual phenotypes for maturity and interactively drag a subset of pods (highlighted in green) in order to group them into 3 clusters according to the desired phenotype categories, shown in Figure 1(a). We hypothesized that, through this interaction, the underlying model would learn new weights for the feature space that satisfy the newly defined projection and properly capture the user's mental model of pod maturity.

Figure 1(b) shows the updated projection (generated after approximately 25 seconds), which produced three main clusters of pods according to their maturity stage. Figure 1(c) shows the ground truth of the images. This indicates that the desired phenotypes were effectively captured by the weighted features and represented in the updated model.

The explainable feature visualizations of specific pods depict the most important visual features learned by the interactive model. In Figure 1(d) we see that one of the important visual features learned by the model to determine the disease phenotype is a salient discolored spot. Similarly, in Figure 1(e,f), the model focuses on image areas correlated to important features of each pod. This provides insight into t parts of the pod are important for visually discerning the maturity stage. Furthermore, these results provide a link between human perception and machine learning.

Number of Pods For the same pods dataset, we also want to explore a different visual phenotype: number of seeds per pod. However, the images were not originally collected to determine the number of seeds. Thus, the number of seeds is a novel visual feature that can be observed directly by the end users but is not initially used to cluster images in the default projection. As before, the images of edamame pods are displayed in the 2D plot. We then interactively drag pods (highlighted in green) to group them into 3 clusters according to the number of seeds (1,2 or 3), as shown in Figure 4(a). We hypothesize that by dragging a subset of the images, the underlying model will learn the weights for the feature spaces that satisfy the user-defined projection based on the number of seeds.

Figure 4(b) shows the updated projection. We find that the projection model captures "number of seeds" phenotype. Figure 4(c) shows the ground truth of the updated projection, instead of well-separated groups, the updated projection shows a linear relationship. We notice that there are two "three-seed" pods projected closer to the "two-seeds" pods. To learn more about why these two pods are mis-projected, we explore the visual feature explanations for each group. Figure 4(d,e,f) shows the saliency map for the three groups accordingly. We find that the most important CNN features mainly capture the overall

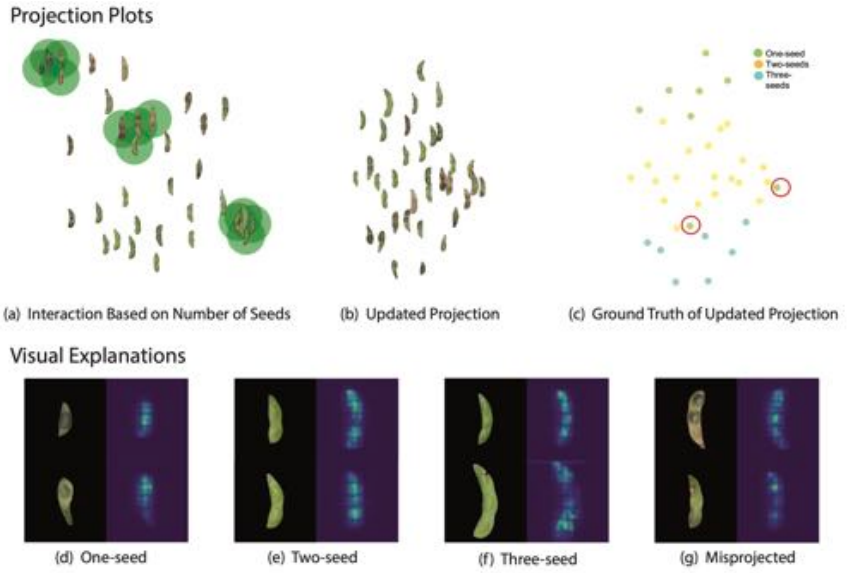

Fig. 4. Interactions to explore images based on the number of seeds. (a) shows the interaction based on seed count. (b) shows the updated projection while (c) shows the ground truth seed count. (d) - (f) shows the explanations of important image features for each seed count while (g) shows the explanations of two misprojected images.

shape of the pod, as well as the position and the "raised" area of the seeds to differentiate pods with different numbers of seeds. Yet for those two mis-projected pods, they are either dominated by the disease spot or do not have the obvious shape of three seeded pods, as shown by Figure 4(g).

6 Discussion

General Framework for Analysis Using Deep Learning Features One of the central problems with using deep learning feature representations in data analysis is the loss of access to the original data features. Typically, people must sacrifice analysis transparency for performance. However, our method presents a framework in which we maintain access to the original data features by leveraging the underlying deep learning model to create explanations from the underlying data features. Through the use of weighted backpropagation, we push the information learned by the projection model back through the neural network to generate explanations relative to the underlying data features. In doing so, we take a step towards solving the "two black boxes" problem, as defined by Wenskovitch and North [34]. The "two black boxes" problem identifies both the deep learning algorithm and the human cognitive process as black boxes that impede the learning process. In our method, semantic interactions with the projection allow people to express some of their cognitive processes to the machine. In return, the model presents explanations that illustrate how it uses the provided information. This creates a

synergy between the machine and the human and facilitates a more complete analysis experience. This framework can be generally applied to analytics methods using deep learning representations of data.

Feature Representation Choice In our method, we use ResNet18 to extract image features. However, alternative methods for feature extraction could be used. Bian et al. explored additional methods for feature extraction, including color histogram and Scale-Invariant Feature Transform [3]. We explored these methods as well but found that feature representations from convolutional neural networks provide the most meaningful projections and explanations. However, there exist other neural network feature extractors besides ResNet18. The design of our method easily allows people to swap in different CNN feature extractors, including those designed for specific tasks and datasets. This allows people to further customize projections of their data for the given analysis task. Additionally, our method can facilitate the comparison of different feature representations to identify the one most appropriate for a given task.

Other Methods for Explanation Our method uses weighted backpropagation to create explanations of the effects of semantic interactions. However, this method is only one candidate for creating explanations of interactions. There exist other methods for generating feature explanations that we can adapt to our method. For example, we also adapted Grad-CAM to consider the weights from the projection model to generate explanations [30]. We found that Grad-CAM excels when images contained multiple entities, however, it falls flat when searching for specific image features. As our method benefits from finer-grained explanations, Grad-CAM was not a suitable method. Adapting other methods for creating model explanations remains to be explored in future work.

7 Conclusion

In this paper, we presented an interactive dimension reduction method for exploring image data using deep learning representations of images. Our method provides semantic interactions that allow people to incorporate their prior knowledge into the projection model. It uses custom-defined relationships to learn new projection weights optimal for respecting these relationships. Additionally, our method provides visual explanations of the effects of semantic interactions on the projections placement of images. These explanations illustrate the image features most important for projecting the images and illustrate the effects of interactions. We provide a real world usage scenario to demonstrate the method's effectiveness at organizing data from human-defined similarities. Overall, we found that our method was able to capture human feedback and incorporate it into the model. Our visual explanations help bridge the gap between the feature space and the original images to illustrate the knowledge learned by the model, creating a synergy between human and machine that facilitates a more complete analysis experience.

Acknowledgements This material is based upon work supported by the National Science Foundation under Grant # 2127309 to the Computing Research Association for the CIFellows 2021 Project. This project was funded, in part, with an integrated internal competitive grant from the College of Agriculture and Life Sciences at Virginia Tech.

References

1. Bian, Y., North, C.: Deepsi: Interactive deep learning for semantic interaction. In: 26th International Conference on Intelligent User Interfaces. pp. 197–207 (2021)
2. Bian, Y., North, C., Krokos, E., Joseph, S.: Semantic explanation of interactive dimensionality reduction. In: 2021 IEEE Visualization Conference (VIS). pp. 26–30. IEEE (2021)
3. Bian, Y., Wenskovitch, J., North, C.: Deepva: Bridging cognition and computation through semantic interaction and deep learning. arXiv preprint arXiv:2007.15800 (2020)
4. Bojarski, M., Choromanska, A., Choromanski, K., Firner, B., Jackel, L., Muller, U., Zieba, K.: Visualbackprop: efficient visualization of cnns. arXiv preprint arXiv:1611.05418 (2016)
5. Brown, E.T., Liu, J., Brodley, C.E., Chang, R.: Dis-function: Learning distance functions interactively. In: 2012 IEEE conference on visual analytics science and technology. pp. 83–92. IEEE (2012)
6. Cavallo, M., Demiralp, Ç.: A visual interaction framework for dimensionality reduction based data exploration. In: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems. pp. 1–13 (2018)
7. Cheng, T.Y., Huertas-Company, M., Conselice, C.J., Aragon-Salamanca, A., Robertson, B.E., Ramachandra, N.: Beyond the hubble sequence–exploring galaxy morphology with unsupervised machine learning. Monthly Notices of the Royal Astronomical Society **503**(3), 4446–4465 (2021)
8. Cunningham, P.: Dimension reduction. In: Machine learning techniques for multimedia, pp. 91–112. Springer (2008)
9. Dowling, M., Wenskovitch, J., Hauck, P., Binford, A., Polys, N., North, C.: A bidirectional pipeline for semantic interaction. In: Proc. Workshop on Machine Learning from User Interaction for Visualization and Analytics (at IEEE VIS 2018). vol. 11, p. 74 (2018)
10. Dowling, M., Wycoff, N., Mayer, B., Wenskovitch, J., House, L., Polys, N., North, C., Hauck, P.: Interactive visual analytics for sensemaking with big text. Big Data Research **16**, 49–58 (2019)
11. Eler, D.M., Nakazaki, M.Y., Paulovich, F.V., Santos, D.P., Andery, G.F., Oliveira, M.C.F., Batista Neto, J., Minghim, R.: Visual analysis of image collections. The Visual Computer **25**(10), 923–937 (2009)
12. Endert, A., Chang, R., North, C., Zhou, M.: Semantic interaction: Coupling cognition and computation through usable interactive analytics. IEEE Computer Graphics and Applications **35**(4), 94–99 (2015)
13. Endert, A., Fiaux, P., North, C.: Semantic interaction for sensemaking: inferring analytical reasoning for model steering. IEEE Transactions on Visualization and Computer Graphics **18**(12), 2879–2888 (2012)
14. Endert, A., Fiaux, P., North, C.: Semantic interaction for visual text analytics. In: Proc. of the SIGCHI Conference on Human Factors in Computing Systems. p. 473–482. CHI '12, ACM, New York, NY, USA (2012). https://doi.org/10.1145/2207676.2207741, https://doi.org/10.1145/2207676.2207741
15. Endert, A., Han, C., Maiti, D., House, L., North, C.: Observation-level interaction with statistical models for visual analytics. In: 2011 IEEE conference on visual analytics science and technology. pp. 121–130. IEEE (2011)
16. Espadoto, M., Appleby, G., Suh, A., Cashman, D., Li, M., Scheidegger, C.E., Anderson, E.W., Chang, R., Telea, A.C.: Unprojection: Leveraging inverse-projections for visual analytics of high-dimensional data. IEEE Transactions on Visualization and Computer Graphics (2021)
17. Fujiwara, T., Wei, X., Zhao, J., Ma, K.L.: Interactive dimensionality reduction for comparative analysis. IEEE Transactions on Visualization and Computer Graphics **28**(1), 758–768 (2022). https://doi.org/10.1109/TVCG.2021.3114807

18. Ghosh, S.K., Biswas, B., Ghosh, A.: A novel noise removal technique influenced by deep convolutional autoencoders on mammograms. In: Deep Learning in Data Analytics, pp. 25–43. Springer (2022)
19. Han, H., Prabhu, R., Smith, T., Dhakal, K., Wei, X., Li, S., North, C.: Interactive deep learning for exploratory sorting of plantimages by visual phenotypes (2022)
20. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
21. House, L., Leman, S., Han, C.: Bayesian visual analytics: Bava. Statistical Analysis and Data Mining: The ASA Data Science Journal **8**(1), 1–13 (2015)
22. Jeong, D.H., Ziemkiewicz, C., Fisher, B., Ribarsky, W., Chang, R.: ipca: An interactive system for pca-based visual analytics. In: Computer Graphics Forum. vol. 28, pp. 767–774. Wiley Online Library (2009)
23. Joia, P., Coimbra, D., Cuminato, J.A., Paulovich, F.V., Nonato, L.G.: Local affine multidimensional projection. IEEE Transactions on Visualization and Computer Graphics **17**(12), 2563–2571 (2011)
24. Leman, S.C., House, L., Maiti, D., Endert, A., North, C.: Visual to parametric interaction (v2pi). PloS one **8**(3), e50474 (2013)
25. Mamani, G.M., Fatore, F.M., Nonato, L.G., Paulovich, F.V.: User-driven feature space transformation. In: Computer Graphics Forum. vol. 32, pp. 291–299. Wiley Online Library (2013)
26. Paulovich, F.V., Eler, D.M., Poco, J., Botha, C.P., Minghim, R., Nonato, L.G.: Piece wise laplacian-based projection for interactive data exploration and organization. In: Computer Graphics Forum. vol. 30, pp. 1091–1100. Wiley Online Library (2011)
27. Pirolli, P., Card, S.: The sensemaking process and leverage points for analyst technology as identified through cognitive task analysis. In: Proceedings of international conference on intelligence analysis. vol. 5, pp. 2–4. McLean, VA, USA (2005)
28. dos Santos Amorim, E.P., Brazil, E.V., Daniels, J., Joia, P., Nonato, L.G., Sousa, M.C.: ilamp: Exploring high-dimensional spacing through backward multidimensional projection. In: 2012 IEEE Conference on Visual Analytics Science and Technology. pp. 53–62. IEEE (2012)
29. Self, J.Z., Dowling, M., Wenskovitch, J., Crandell, I., Wang, M., House, L., Leman, S., North, C.: Observation-level and parametric interaction for high-dimensional data analysis. ACM Transactions on Interactive Intelligent Systems (TiiS) **8**(2), 1–36 (2018)
30. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017)
31. Tukey, J.W., Wilk, M.B.: Data analysis and statistics: an expository overview. In: Proceedings of the November 7-10, 1966, fall joint computer conference. pp. 695–709 (1966)
32. Villaret, M., et al.: Affective state-based framework for e-learning systems. In: Artificial Intelligence Research and Development: Proceedings of the 23rd International Conference of the Catalan Association for Artificial Intelligence. vol. 339, p. 357. IOS Press (2021)
33. Wang, M., Wenskovitch, J., House, L., Polys, N., North, C.: Bridging cognitive gaps between user and model in interactive dimension reduction. Visual Informatics **5**(2), 13–25 (2021)
34. Wenskovitch, J., North, C.: Interactive ai: Designing for the 'two black boxes' problem. Hybrid Human-Artificial Intelligence Special Issue (Washington, United States: IEEE Computer Society pp. 1–10 (2020)
35. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: European conference on computer vision. pp. 818–833. Springer (2014)
36. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2921–2929 (2016)