VisLRPDesigner Team

User Guide of VisLRPDesigner

Contents

Foreword

Layer-wise Relevance Propagation (LRP) methods are widely used in the explanation of deep neural networks (DNN), especially in computer vision field for interpreting the prediction results of convolutional neural networks (CNN). Multiple LRP variations utilize a set of relevance backpropagation rules with various parameters. Moreover, composite LRPs apply different rules on segments of CNN layers. These features impose great challenge for users to design, explore, and find suitable LRP models. VisLRPDesigner is a Web-based visualization software which is designed to help LRP designers and students efficiently perform these tasks. Various LRPs are unified into an integrated framework with an intuitive workflow of parameter setup. Therefore, VisLRPDesigner allows users to interactively configure LRP models, change parameters, and then study the relevance information. VisLRPDesigner further facilitates relevance based visual analysis with two functions: relevance-based pixel flipping and neuron ablation.

VisLRPDesigner software, documents, and demo videos are a Web-based software which can be accessed at http://vis.cs.kent.edu/VisLRPDesigner. The current version is based on the CNN VGG16 model only.

1

Overview of Layer-wise Relevance Propagation and VisLRPDesigner

CONTENTS

1.1 Layer-wise Relevance Propagation for Deep Learning Explanation

LRP techniques explain the prediction of a deep learning model, such as convolutional neural network (CNN), by finding the relevance of input image pixels to the output [1]. By initiating relevance based on a selected output class, a backward propagation from the output layer to the lower layers is employed to compute relevance values at each layer and towards the input pixels. Each neuron receives a share of the relevance values from its successor nodes, and redistributes its relevance to predecessor neurons while the conservation of relevance is ensured. During this backprogagation process, the distribution of relevance can be computed by using different relevance propagation rules (i.e., functions) that utilize the forward neuron activations and a set of artificial parameters. Composite LRPs further allow different propagation rules and parameters being used at different layers selected by users. With respect to these variations and selections, LRPs create different relevance values which are often shown as heatmaps to explain their contribution towards CNN output. Please see an overview [5] of LRP for more details.

LRP computational process is illustrated in Figure 1.1. The relevance computation is implemented in two phases:

- First, a standard forward propagation pass is applied to the network from an input image. The activation a_i of each neuron x_i^l at layer l is collected. The network weight from neuron x_i^l to neuron x_j^{l+1} in its successor layer $l+1$ is also recorded as w_{ij}.

- Second, with a layer-wise backward propagation pass, a relevance map R_k^l is computed to represent the relevance of each neuron k at each layer l. The computation starts from an initial (input) relevance vector R defined at the output layer. Then a backpropagation from layer $l+1$ to layer l is implemented as:

  $$R_i^l = \sum_j \mathcal{F}(a_i, w_{ij}) R_j^{l+1}, \tag{1.1}$$

  where $\mathcal{F}$ is the LRP propagation rule (function). Meanwhile, the conservation of relevance is ensured by

  $$\sum_i R_i^l = \sum_j R_j^{l+1}. \tag{1.2}$$

  This process stops when a *relevance map* R^0 on the input image (i.e., layer $l = 0$) is achieved.

The result relevance map is shown as a heatmap of the imput image, where red pixels indicates high positive relevance and blue refers to high negative relevance.

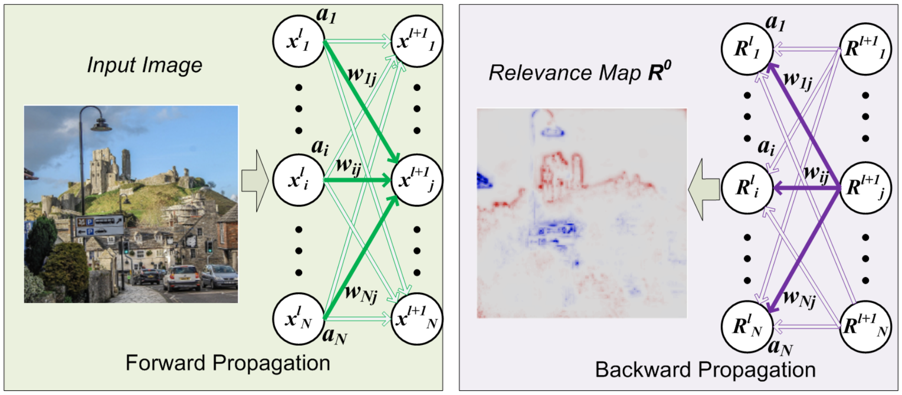


FIGURE 1.1
LRP computational process in the neural network and the result of the relevance heatmap.

1.2 LRP Backpropagation Rules

In the original paper of LRP [1], the propagation rule $\mathcal{F}$ is defined in two popular forms called $LRP - \epsilon$ and $LRP - \alpha\beta$:

$$LRP\text{-}\epsilon: \qquad R_i^l = \sum_j \frac{a_i w_{ij}}{\epsilon + \sum_i a_i w_{ij}} R_j^{l+1}, \tag{1.3}$$

where a small constant ϵ prevents the numerical instability when the denominator becomes zero. The rule without ϵ is called $LRP - 0$.

$$LRP\text{-}\alpha\beta: \qquad R_i^l = \sum_j (\alpha \cdot \frac{a_i w_{ij}^+}{\sum_i a_i w_{ij}^+} - \beta \cdot \frac{a_i w_{ij}^-}{\sum_i a_i w_{ij}^-}) R_j^{l+1}, \tag{1.4}$$

where $()^+$ and $()^-$ denote the positive weights and the negative weights, and α and β are chosen parameters with $\alpha - \beta = 1$. $LRP - \alpha\beta$ splits the positive and negative activations in the relevance propagation process. Using different parameters α and β can modulate the qualitative behavior of the resulting explanation with different contributions of excitatory and inhibitory effects (see [7]). The result LRPs are usually referred by the α and β values, such as $LRP - \alpha 2 \beta 1$ for using $\alpha = 2$ and $\beta = 1$, and $LRP - \alpha 1 \beta 0$ for using $\alpha = 1$ and $\beta = 0$.

Researchers have further proposed more LRP forms by manipulating the propagation function $\mathcal{F}$, such as LRP-γ, LRP-w^2, LRP-z^+, LRP-z^-, and so on. For example, one popular enhancement approach is LRP-γ which emphasizes on positive contributions (controlled by the parameter γ) over negative

TABLE 1.1
Multiple LRP rules rules and their properties.[5]

LRP Rule	**Formula**	**Usage**	**Reference**
LRP-0	$\mathrm{R}_i = \Sigma_j \frac{a_i w_{ij}}{\Sigma_i a_i w_{ij}} R_j$	Upper layers	[1]
LRP-ϵ	$\mathrm{R}_i = \Sigma_j \frac{a_i w_{ij}}{\epsilon + \Sigma_i a_i w_{ij}} R_j$	Middle layers	[1]
LRP-γ	$\mathrm{R}_i = \Sigma_j \frac{a_i (w_{ij} + \gamma w_{ij}^{+})}{\Sigma_i a_i (w_{ij} + \gamma w_{ij}^{+})} R_j$	Lower layers	[5]
LRP-$\alpha\beta$	$\mathrm{R}_i = \Sigma_j (\alpha \frac{a_i w_{ij}^{+}}{\Sigma_i a_i w_{ij}^{+}} - \beta \frac{a_i w_{ij}^{-}}{\Sigma_i a_i w_{ij}^{-}}) R_j$	Lower layers	[1]
LRP-z^+	$\mathrm{R}_i = \Sigma_j \frac{a_i w_{ij}^{+}}{\Sigma_i a_i w_{ij}^{+}} R_j$	Lower layers	[1]
LRP-flat	$\mathrm{R}_i = \Sigma_j \frac{1}{\Sigma_i 1} R_j$	Lower layers	[3]
LRP-z^β	$\mathrm{R}_i = \Sigma_j \frac{a_i w_{ij} - l_i w_{ij}^{+} - h_i w_{ij}^{-}}{\Sigma_i a_i w_{ij} - l_i w_{ij}^{+} - h_i w_{ij}^{-}} R_j$	First layer (pixels)	[6]
LRP-w^2	$\mathrm{R}_i = \Sigma_j \frac{w_{ij}^2}{\Sigma_i w_{ij}^2} R_j$	First layer (R^d)	[6]

contributions as:

$$LRP\text{-}\gamma: \qquad R_i^l = \sum_j \frac{a_i (w_{ij} + \gamma w_{ij}^{+})}{\sum_i a_i (w_{ij} + \gamma w_{ij}^{+})} R_j^{l+1}. \tag{1.5}$$

In Table 2.1, a set of popularly used LRP rules from existing literature are summarized with their backpropagation functions. These variations lead to different performances on CNN models but there is no best solution for all cases. For example, as discussed in [7], $LRP - \alpha 2 \beta 1$ is shown [1] to work well on BVLC CaffeNet [2], and $LRP - \alpha 1 \beta 0$ is discovered [6] to be more stable for GoogleNet [10].

Moreover, a *composite LRP* strategy is proposed [8], where different rules are used at different layers. It provides more flexibility for users to decide and leverage the benefits of different LRP rules. For instance, it has been suggested that $LRP - 0$ for top layers, $LRP - \epsilon$ for middle layers, and $LRP - \gamma$ for lower layers [5].

Figure. 1.2 shows an example of the composite LRP where different rules are applied on multiple segments of CNN layers.

1.3 LRP Design Challenges

The flexibility of LRP imposes great burden for users to apply LRP successfully. In practice, a trial-and-error process often needs to be conducted to investigate different propagation rules and parameter settings. Composite LRPs further confound the situation when various rules are tested on different CNN layers. Notice users are often perplexed in understanding and manipulating

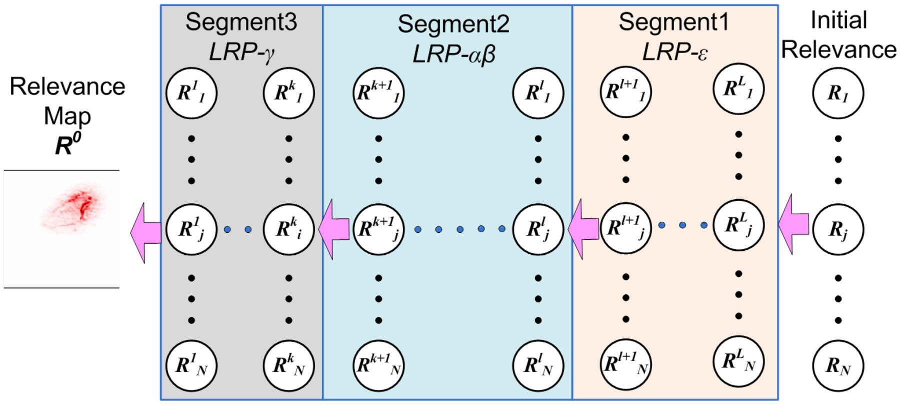


FIGURE 1.2
An example of composite LRP rules in the segments of a CNN.

this process. Even experienced users need to spend great time and effort in model design, validation, adjustment, and comparison. Unfortunately, in most cases, this exploratory process has to be implemented manually in the coding stage.

1.4 Existing LRP Design Tools

Coding method: One LRP tutorial is implemented in pytorch which can be found by this URL, http://www.heatmapping.org/tutorial/. The other LRP library, Innvestigate, is also provided in tensorflow, which can be found in this github https://github.com/albermax/innvestigate. However, it takes time for users to get familiar with the library and it may be not convenient for users to compare the different results.
Online demo: An interactive demo interface is also provided to study the LRP in this link https://lrpserver.hhi.fraunhofer.de/image-classification. This interactive interface demo is convenient to all levels of users for knowing the basic concept of LRP, since some parameters can be selected by users. But it is still limited to fully control all the parameters of a LRP configuration in a neural network model, especially for the composite LRP formula. The layer range for those composite rules are fixed, and this user interface doesn't allow users to make more changes on the rules in different ranges of layers. Also, the details of the composite LRP method are not shown too.

1.5 Goal of VisLRPDesigner

VisLRPDesigner is a visualization software which facilitates easy and intuitive design of LRP models through a Web-based interactive design interface. It is developed to further improve the usability of LRP and enhance the understanding of LRP techniques, so as to promote wider use in deep learning explanation. The main features and contributions of this software are as follows:

1.6 Features of VisLRPDesigner

VisLRPDesigner is designed to facilitate LRP researchers and users with the following functions.

Interactive LRP Model Design:

- Users can easily configure preferred LRP rules on an LRP configuration panel through four visually interactive parameter-setting steps.
- Users can define and revise composite LRP model by drag-and-define multiple segments of CNN layers, and apply different LRP rules over these segments.
- Users can immediately and visually examine the LRP computation results (relevance maps) during their interactive design of LRP models. They can perform the test over different images easily.
- Users can manage their created LRP models by saving and loading them in the software.
- Users can compare multiples models with their LRP relevance maps.

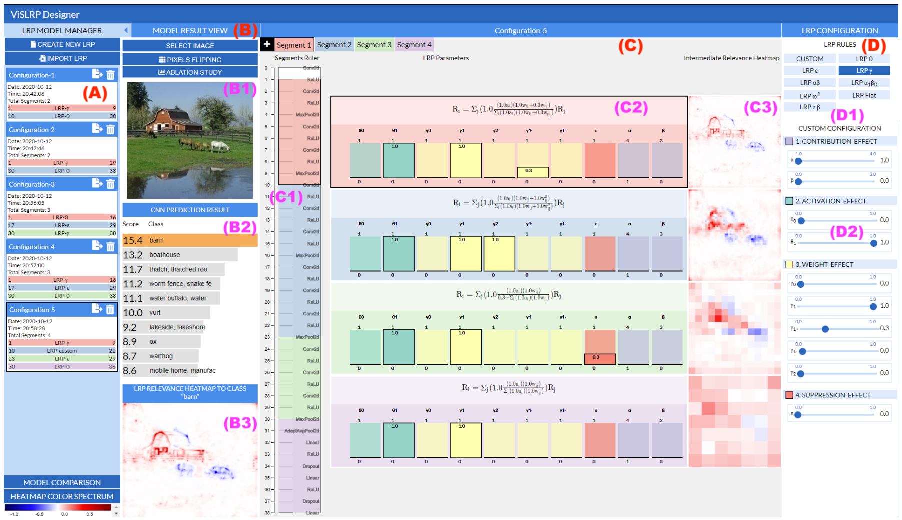
Visual Model Analysis:

- Users can perform relevance-based pixel flipping to study the change of the CNN prediction after flipping high- or low-relevance pixels to zero in the input image.
- Users can interactively remove specific neurons in the CNN based on LRP relevance scores, so as to evaluate how the prediction results change with specific neurons.

2

VisLRPDesigner Interface and Functions

CONTENTS



Configuration-1
Date: 2020-10-12
Time: 20:42:08
Total Segments: 2
1 LRP-γ 9
10 LRP-0 38

Configuration-2
Date: 2020-10-12
Time: 20:42:46
Total Segments: 2
1 LRP-γ 29
30 LRP-0 38

Configuration-3
Date: 2020-10-12
Time: 20:56:05
Total Segments: 3
1 LRP-0 16
17 LRP-ε 29
30 LRP-γ 38

Configuration-4
Date: 2020-10-12
Time: 20:57:00
Total Segments: 3
1 LRP-γ 16
17 LRP-ε 29
30 LRP-0 38

Configuration-5
Date: 2020-10-12
Time: 20:58:28
Total Segments: 4
1 LRP-γ 9
10 LRP-custom 22
23 LRP-ε 29
30 LRP-0 38

Score	Class
15.4	barn
13.2	boathouse
11.7	thatch, thatched roo
11.2	worm fence, snake fe
11.1	water buffalo, water
10.0	yurt
9.2	lakeside, lakeshore
8.9	ox
8.7	warthog
8.6	mobile home, manufac

1. CONTRIBUTION EFFECT: α 1.0; β 0.0
2. ACTIVATION EFFECT: θ_0 0.0; θ_1 1.0
3. WEIGHT EFFECT: γ_0 0.0; γ_1 1.0; γ_{1+} 0.3; γ_{1-} 0.0; γ_2 0.0
4. SUPPRESSION EFFECT: ε 0.0

FIGURE 2.1
Overview of VisLRPDesigner interface.

2.1 Overview of VisLRPDesigner Interface

VisLRPDesigner interface contains four major panels as shown in Figure 2.1:

- (A): Model manager for users to create, edit, and compare multiple LRP models, and to change heatmap color spectrum.
- (B): Model result view for selecting image of interest (B1), checking CNN prediction results (B2), and studying LRP computed relevance heatmap (B3).
- (C): LRP configuration view which includes a segment ruler (C1) for users to drag and define segments, an LRP parameter view (C2) visualizing parameters of four segments, and intermediate relevance heatmaps (C3) after each segment.
- (D): LRP configuration panel for users to select a predefined LRP rule (D1) and customize LRP parameters (D2).

The system also contains a few sample images as shown in Figure 2.2, which can be selected by clicking the SELECT IMAGE button in panel (B). After an image is selected, its vgg16 [9] prediction results are shown in (B1) and (B2).

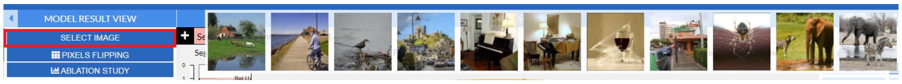


FIGURE 2.2
Image Selection.

2.2 LRP Model Configuration

VisLRPDesigner support model configuration with the functions:

- Setting segments of layers for composite LRP
- Defining LRP Rules
- Running LRP models
- Managing LRP Models

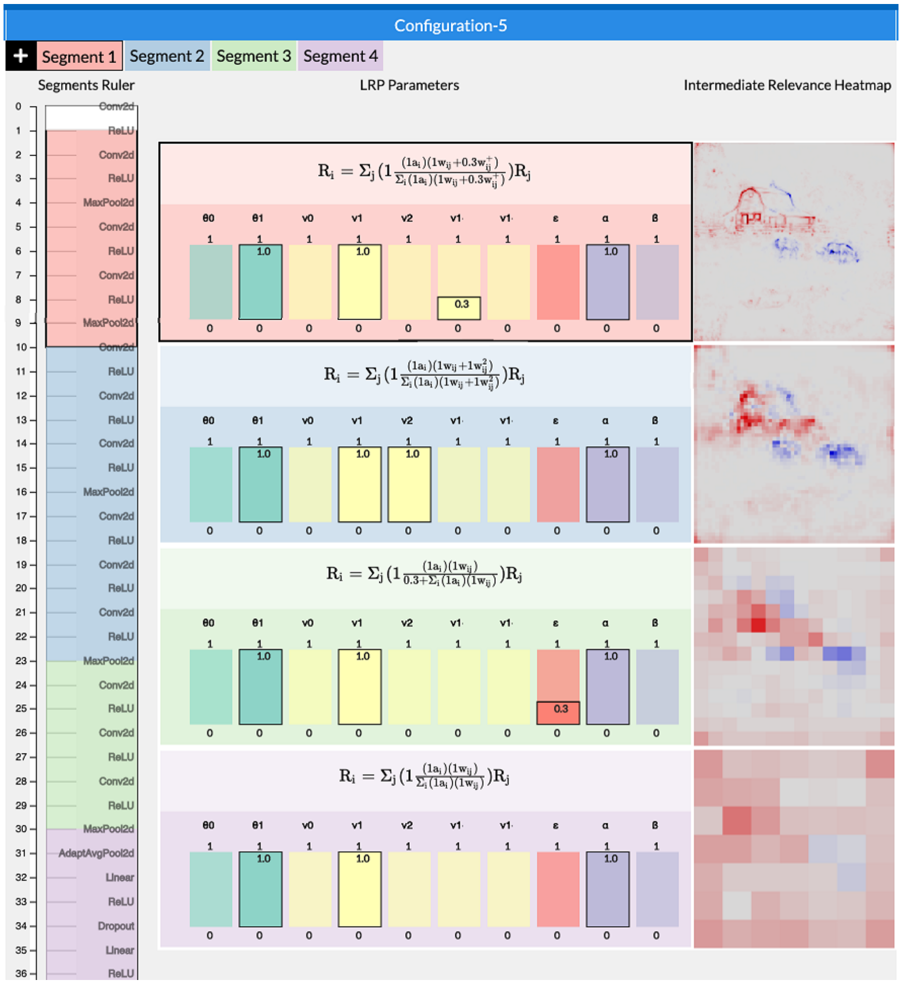
2.2.1 Setting Segments of Layers for Composite LRP

Since a deep neural network model contains multiple layers, and the roles of each layer in different ranges of depth are different, different rules can be applied accordingly on the deep neural network model in a composite LRP model. In the implementation, the hidden layers can be grouped into segments, and each segment uses specific LRP rule as shown in Figure 2.4.

The figure shows the vgg16 model with 39 layers. The layers are visualized in a ruler at the left side, where 4 segments are highlighted in different colors and each segment is applied with a different rule as shown by the formula above the parameter bar charts. The intermediate relevance heatmaps of these segments are listed at the right column, which show the relevance distribution of the hidden layers at the end of each segment.

For segments setting, by default, the basic model with LRP-0 in all layers will be initialized as a base model in the configuration interface for segments setting first. The configuration information of this base model and any changes on it will also be synchronized and saved to a temporal model unit named as “My New Model” marked in Figure 2.2.1, which will be discussed about managing this temporary modal in model management Section 2.2.4. Next, we show the steps of setting segments.

Step 1: Click “+” button on the top left corner of the segment configuration panel. A new segment will show up at the bottom of the ruler.



$$R_i = \Sigma_j (1 \frac{(1a_i)(1w_{ij}+0.3w_{ij}^{+})}{\Sigma_i (1a_i)(1w_{ij}+0.3w_{ij}^{+})})R_j$$

θ0	θ1	v0	v1	v2	v1·	v1·	ε	α	β
1	1	1	1	1	1	1	1	1	1
	1.0		1.0		0.3			1.0	
0	0	0	0	0	0	0	0	0	0

$$R_i = \Sigma_j (1 \frac{(1a_i)(1w_{ij}+1w_{ij}^{2})}{\Sigma_i (1a_i)(1w_{ij}+1w_{ij}^{2})})R_j$$

θ0	θ1	v0	v1	v2	v1·	v1·	ε	α	β
1	1	1	1	1	1	1	1	1	1
	1.0		1.0	1.0				1.0	
0	0	0	0	0	0	0	0	0	0

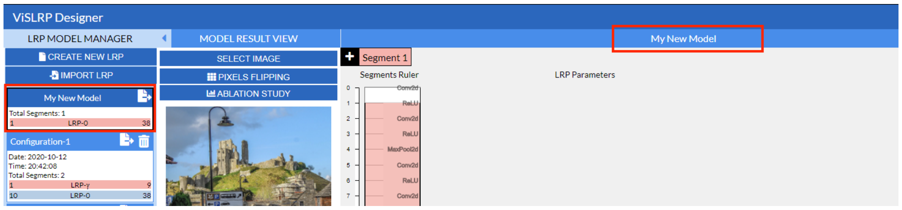
$$R_i = \Sigma_j (1 \frac{(1a_i)(1w_{ij})}{0.3+\Sigma_i (1a_i)(1w_{ij})})R_j$$

θ0	θ1	v0	v1	v2	v1·	v1·	ε	α	β
1	1	1	1	1	1	1	1	1	1
	1.0		1.0				0.3	1.0	
0	0	0	0	0	0	0	0	0	0

$$R_i = \Sigma_j (1 \frac{(1a_i)(1w_{ij})}{\Sigma_i (1a_i)(1w_{ij})})R_j$$

θ0	θ1	v0	v1	v2	v1·	v1·	ε	α	β
1	1	1	1	1	1	1	1	1	1
	1.0		1.0					1.0	
0	0	0	0	0	0	0	0	0	0

FIGURE 2.3
Visual interface of a composite LRP with four segments.



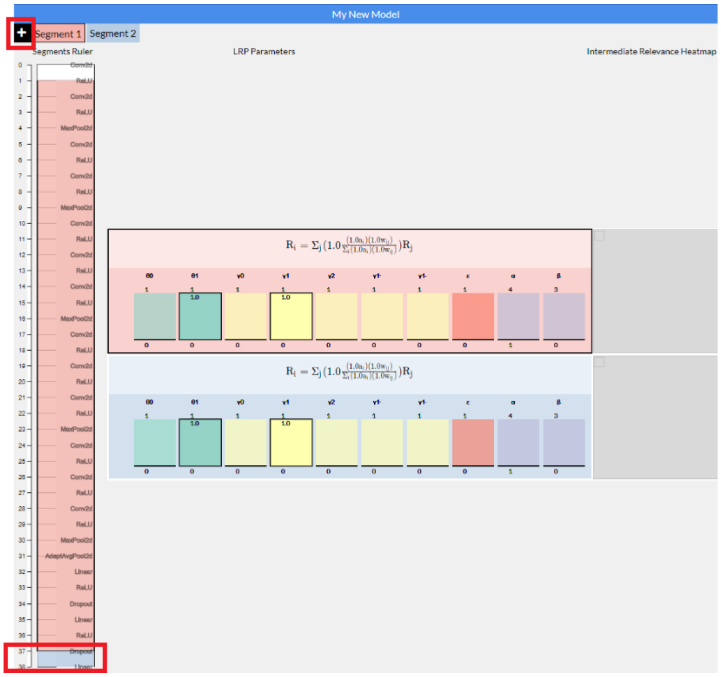
My New Model
Total Segments: 1

1	LRP-0	38

Configuration-1
Date: 2020-10-12
Time: 20:42:08
Total Segments: 2

1	LRP-γ	9
10	LRP-0	38

FIGURE 2.4
Initialized "My New Model" for model design.



Step 2: Drag the border line between two segments to cover the preferred layers of the segments.
Step 3: Define the formula of any segment by clicking the segment, and then set its LRP rule with LRP configuration panel (see below).

2.2.2 Defining LRP Rules

LRP rules and workflow

Most existing LRP rules can be unified into one formula:

$$R_i^l = \sum_j \Bigg[\alpha \frac{(\theta_0+\theta_1 a_i)(\gamma_0+\gamma_1 w_{ij}+\gamma_{1p}w_{ij}^+ +\gamma_{1n}w_{ij}^- +\gamma_2 w_{ij}^2)}{\epsilon+\Sigma_i(\theta_0+\theta_1 a_i)(\gamma_0+\gamma_1 w_{ij}+\gamma_{1p}w_{ij}^+ +\gamma_{1n}w_{ij}^- +\gamma_2 w_{ij}^2)}$$
$$-\beta\frac{(\theta_0+\theta_1 a_i)(\gamma_0+\gamma_1 w_{ij}+\gamma_{1p}w_{ij}^+ +\gamma_{1n}w_{ij}^- +\gamma_2 w_{ij}^2)}{\epsilon+\Sigma_i(\theta_0+\theta_1 a_i)(\gamma_0+\gamma_1 w_{ij}+\gamma_{1p}w_{ij}^+ +\gamma_{1n}w_{ij}^- +\gamma_2 w_{ij}^2)}\Bigg]R_j^{l+1}. \quad (2.1)$$

Various LRP rules can be achieved by using different sets of the ten parameters within this formula. In addition to use heuristic values presented in the literature, VisLRPDesigner allows users to try different values: for example, instead of simply using α and β as either 1 or 0, they can possibly try float values to flexibly combine excitatory and inhibitory effects.

The ten parameters in the formula can be categorized into four groups with respect to their roles:

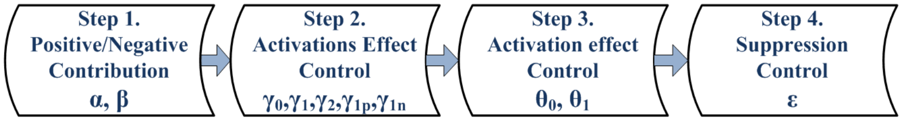


FIGURE 2.5
A 4-step workflow of LRP parameter setting within Eqn 2.1.

TABLE 2.1
Multiple LRP rules and their settings within our unified formula.

Rule	Ref.	Original Formula	Parameters in the Unified Formula (Eqn. 2.1)
LRP-0	[1]	$\mathrm{R}_i = \Sigma_j \frac{a_i w_{ij}}{\Sigma_i a_i w_{ij}} R_j$	$\alpha = 1,\ \beta = 0,\ \theta_0 = 0,\ \theta_1 = 1,\ \gamma_0 = 0,\ \gamma_1 = 1,\ \gamma_{1p} = 0,\ \gamma_{1n} = 0,\ \gamma_2 = 0,\ \epsilon = 0$
LRP-ϵ	[1]	$\mathrm{R}_i = \Sigma_j \frac{a_i w_{ij}}{\epsilon + \Sigma_i a_i w_{ij}} R_j$	$\alpha = 1,\ \beta = 0,\ \theta_0 = 0,\ \theta_1 = 1,\ \gamma_0 = 0,\ \gamma_1 = 1,\ \gamma_{1p} = 0,\ \gamma_{1n} = 0,\ \gamma_2 = 0,\ \epsilon \in [0, 1]$
LRP-γ	[5]	$\mathrm{R}_i = \Sigma_j \frac{a_i (w_{ij} + \gamma w_{ij}^{+})}{\Sigma_i a_i (w_{ij} + \gamma w_{ij}^{+})} R_j$	$\alpha = 1,\ \beta = 0,\ \theta_0 = 0,\ \theta_1 = 1,\ \gamma_0 = 0,\ \gamma_1 = 1,\ \gamma_{1p} = 1,\ \gamma_{1n} = 0,\ \gamma_2 = 0,\ \epsilon = 0$
LRP-$\alpha\beta$	[1]	$\mathrm{R}_i = \Sigma_j (\alpha \frac{a_i w_{ij}^{+}}{\Sigma_i a_i w_{ij}^{+}} - \beta \frac{a_i w_{ij}^{-}}{\Sigma_i a_i w_{ij}^{-}}) R_j$	$\alpha - \beta = 1,\ \beta \leq 0,\ \theta_0 = 0,\ \theta_1 = 1,\ \gamma_0 = 0,\ \gamma_1 = 0,\ \gamma_{1p} = 1(0),\ \gamma_{1n} = 0(1)$ for α (β), $\gamma_2 = 0,\ \epsilon = 0$
LRP-z^{+}	[1]	$\mathrm{R}_i = \Sigma_j \frac{a_i w_{ij}^{+}}{\Sigma_i a_i w_{ij}^{+}} R_j$	$\alpha = 1,\ \beta = 0,\ \theta_0 = 0,\ \theta_1 = 1,\ \gamma_0 = 0,\ \gamma_1 = 0,\ \gamma_{1p} = 1,\ \gamma_{1n} = 0,\ \gamma_2 = 0,\ \epsilon = 0$
LRP-z^{β}	[6]	$\mathrm{R}_i = \Sigma_j \frac{a_i w_{ij} - l_i w_{ij}^{+} - h_i w_{ij}^{-}}{\Sigma_i a_i w_{ij} - l_i w_{ij}^{+} - h_i w_{ij}^{-}} R_j$	$\alpha = 1,\ \beta = 0,\ \theta_0 = 0,\ \theta_1 = 1,\ \gamma_0 = 0,\ \gamma_1 = 1,\ \gamma_{1p} = -l_i,\ \gamma_{1n} = -h_i,\ \gamma_2 = 0,\ \epsilon = 0$
LRP-w^2	[6]	$\mathrm{R}_i = \Sigma_j \frac{w_{ij}^2}{\Sigma_i w_{ij}^2} R_j$	$\alpha = 1,\ \beta = 0,\ \theta_0 = 1,\ \theta_1 = 0,\ \gamma_0 = 0,\ \gamma_1 = 0,\ \gamma_{1p} = 0,\ \gamma_{1n} = 0,\ \gamma_2 = 1,\ \epsilon = 0$
LRP-flat	[3]	$\mathrm{R}_i = \Sigma_j \frac{1}{\Sigma_i 1} R_j$	$\alpha = 1,\ \beta = 0,\ \theta_0 = 1,\ \theta_1 = 0,\ \gamma_0 = 1,\ \gamma_1 = 0,\ \gamma_{1p} = 0,\ \gamma_{1n} = 0,\ \gamma_2 = 0,\ \epsilon = 0$

- **Activation effect control**: θ_0 and θ_1 which determine the effect of the activations recorded in the forward pass. Most LRP rules have $\theta_0 = 0$ and $\theta_1 = 1$, while other rules such as LRP-w^2 do not involve activations, which can be set with $\theta_0 = 1$ and $\theta_1 = 0$.

- **Weight effect control**: γ_0, γ_1, γ_{1p},γ_{1n} and γ_2. They control the contributions from the positive weight, negative weight, and square weight, which are selectively used in different rules.

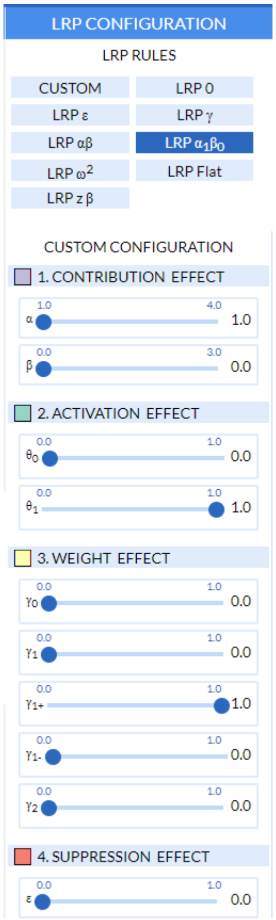
- **Suppression control**: ϵ which is to handle zero division and suppress the background noise.

- **Positive/negative contribution**: α and β which define the final relevance with different contributions from the positive and negative backpropagated relevances.

With the four groups, a parameter setting workflow is enabled as shown in Figure 2.5. Therefore, users can define their preferred LRP rule in a step by step process. In Table 2.1, a set of popularly used LRP rules from existing literature are summarized with their backpropagation functions.

Visual definition process

Users can interactively set LRP rule for one segment in LRP CONFIGURATION panel as shown in Figure 2.6. First, seven popular LRP rules are



1. CONTRIBUTION EFFECT

α (1.0–4.0): 1.0

β (0.0–3.0): 0.0

2. ACTIVATION EFFECT

θ_0 (0.0–1.0): 0.0

θ_1 (0.0–1.0): 1.0

3. WEIGHT EFFECT

γ_0 (0.0–1.0): 0.0

γ_1 (0.0–1.0): 0.0

γ_{1+} (0.0–1.0): 1.0

γ_{1-} (0.0–1.0): 0.0

γ_2 (0.0–1.0): 0.0

4. SUPPRESSION EFFECT

ε (0.0–1.0): 0.0

FIGURE 2.6
LRP Configuration Panel

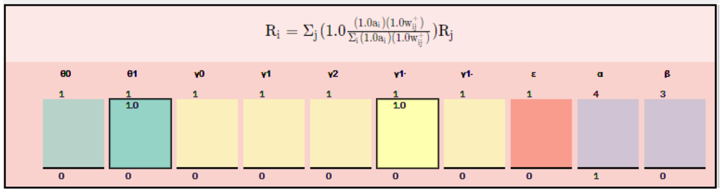
provided as the templates for direct selection. Then, users can adjust parameter values such as α and β.

Users can also directly define a rule by clicking the CUSTOM button and tune the values of parameters. The tuned values for the LRP formula will be updated on the Segment Configuration panel as shown in Figure 2.7.

2.2.3 Running LRP models

Once an LRP model is defined, users can execute LRP backpropagation and generate relevance maps corresponding to different prediction classes.

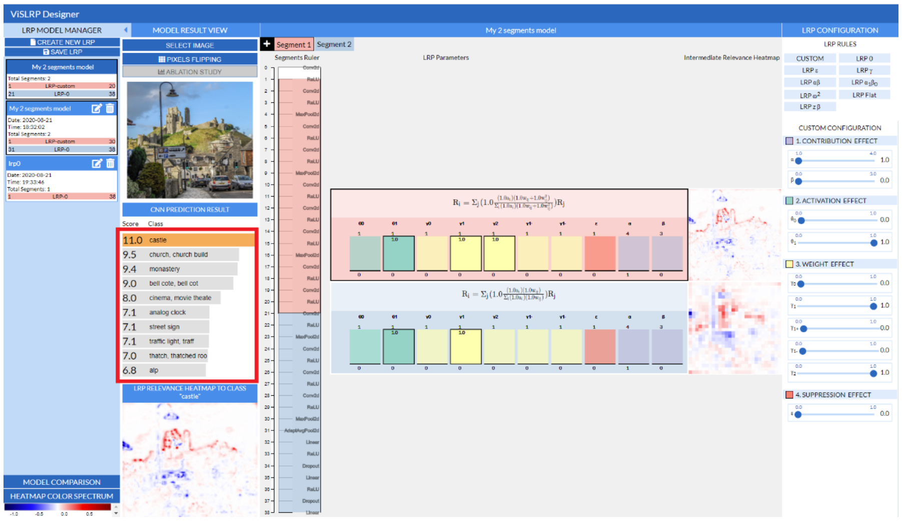
Users can click any class bar on the list of prediction classes. VisLRPDesigner immediately shows the generated relevance heatmap. As illustrated in



$$R_i = \Sigma_j (1.0 \frac{(1.0a_i)(1.0w_{ij}^+)}{\Sigma_i (1.0a_i)(1.0w_{ij}^+)}) R_j$$

θ0	θ1	γ0	γ1	γ2	γ1-	γ1-	ε	α	β
1	1	1	1	1	1	1	1	4	3
	1.0				1.0				
0	0	0	0	0	0	0	0	1	0

FIGURE 2.7
LRP formula and parameter view.



My 2 segments model
Total Segments: 2

1	LRP-custom	20
21	LRP-0	38

My 2 segments model
Date: 2020-08-21
Time: 18:32:02
Total Segments: 2

1	LRP-custom	30
31	LRP-0	38

lrp0
Date: 2020-08-21
Time: 19:33:46
Total Segments: 1

1	LRP-0	38

Score	Class
11.0	castle
9.5	church, church build
9.4	monastery
9.0	bell cote, bell cot
8.0	cinema, movie theate
7.1	analog clock
7.1	street sign
7.1	traffic light, traff
7.0	thatch, thatched roo
6.8	alp

1. CONTRIBUTION EFFECT
α 1.0
β 0.0

2. ACTIVATION EFFECT
θ_0 0.0
θ_1 1.0

3. WEIGHT EFFECT
γ_0 0.0
γ_1 1.0
γ_{1+} 0.0
γ_{1-} 0.0
γ_2 1.0

4. SUPPRESSION EFFECT
ε 0.0

FIGURE 2.8
The castle class is selected as the target class.

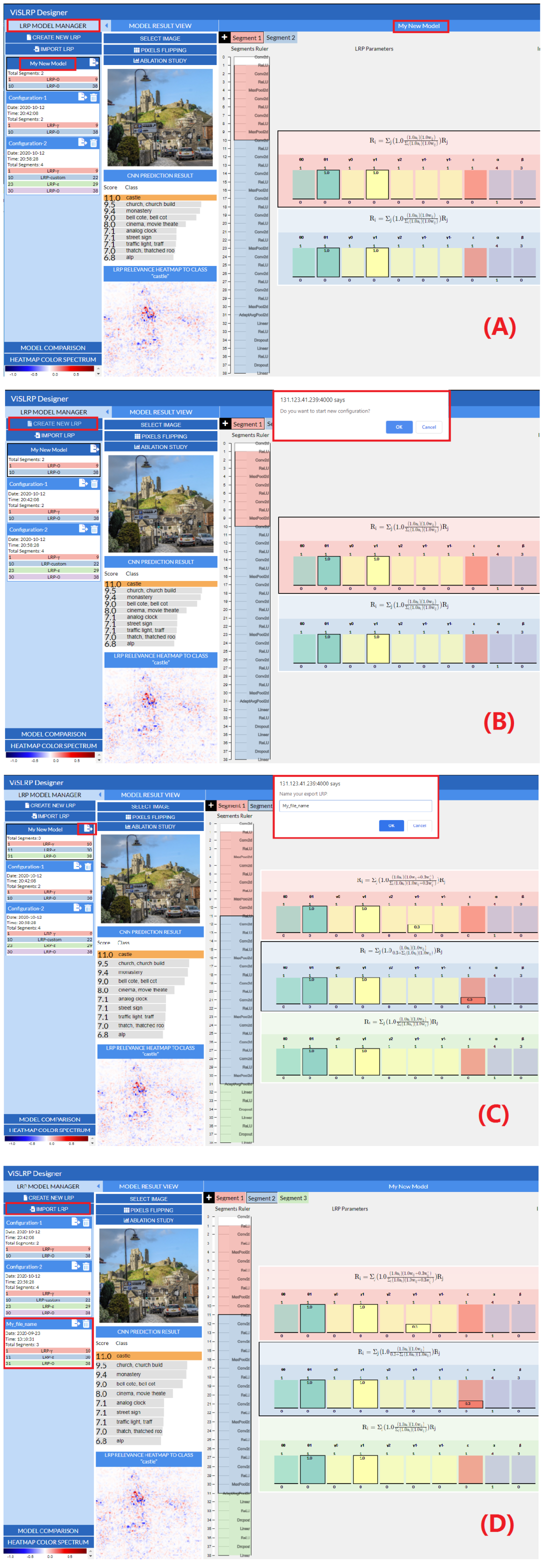
Figure 2.8, the class "castle" is selected and the heatmap is shown in the bottom. Intermediate heatmaps of each LRP segment are also shown.

Users can click on the color spectrum (on the bottom left corner of the interface) to change the colors used to visualize the relevance scores on heatmaps.

2.2.4 Managing LRP models

LRP models can be imported and exported in LRP MODEL MANAGER as shown in Figure 2.9 (A-D).

- "My New Model" is used to save the temporal new designed model on the server in Figure 2.9 (A). Any modification on the interface of segments setting will be updated in the "My New Model" in LRP MODEL MANAGER.



131.123.41.239:4000 says

Do you want to start new configuration?

OK Cancel

131.123.41.239:4000 says

Name your export LRP

My_file_name

OK Cancel

$$R_i = \Sigma_j\left(1.0\frac{(1.0a_i)(1.0w_{ij})}{\Sigma_i(1.0a_i)(1.0w_{ij})}\right)R_j$$

$$R_i = \Sigma_j\left(1.0\frac{(1.0a_i)(1.0w_{ij}+0.3w_{ij}^{+})}{\Sigma_i(1.0a_i)(1.0w_{ij}+0.3w_{ij}^{+})}\right)R_j$$

FIGURE 2.9
LRP Model management.

- To discard the current configuration, we can click the button, CREATE NEW LRP, and click “OK” to refresh the configuration in Figure 2.9 (B).
- The well designed LRP model can be exported to the user’s local directory by clicking the exporting button by the side of “My New Model” and typing a file name for this LRP model in Figure 2.9 (C).
- By clicking the IMPORT LRP MODEL button, users are prompted to upload the json file of LRP model from users’ end. Then, this model will be loaded to LRP MODEL MANAGER in Figure 2.9 (D).

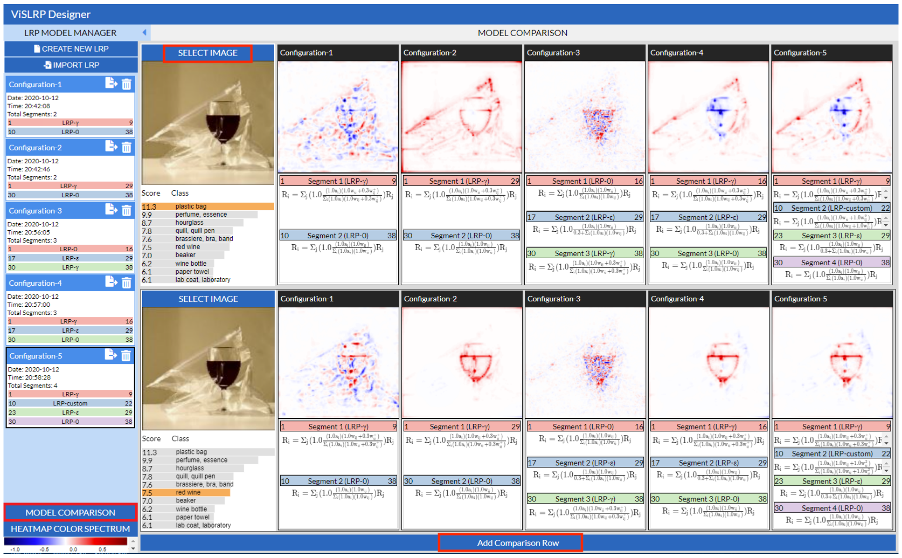
Currently, five LRP model multiple segments are provided as examples. Users can add more models and also delete any model by clicking the trash button for the unnecessary model. From the model list, users also can set any model to be active by clicking on it, and then either use it directly or change its model configuration to perform further study.

2.3 LRP Model Comparison

Different LRP models can be compared and examined visually by clicking the MODEL COMPARISON button at the bottom of LRP MODEL MANAGER panel.

In the model comparison view shown in Figure 2.9, user can perform model comparison in multiple steps:

- Step 1: A selected image with its CNN prediction classes are loaded into one row of this view. They can also change the target image at each row.
- Step 2: Users can select LRP models for comparison in one row. They can also change the target class.
- Step 3: The resultant heatmaps of relevances at each cell with corresponding models are shown together. Meanwhile, the corresponding LRP model rules are shown for each cell.
- Step 4: Users can add more rows with the same image and use different prediction class for relevance computation by clicking **Add Comparison Row** at the bottom of the rows and also the gray area of the new row for activating. They can also change the target image by clicking SELECT IMAGE as well.



Configuration-1
Date: 2020-10-12
Time: 20:42:08
Total Segments: 2

1	LRP-γ	9
10	LRP-0	38

Configuration-2
Date: 2020-10-12
Time: 20:42:46
Total Segments: 2

1	LRP-γ	29
30	LRP-0	38

Configuration-3
Date: 2020-10-12
Time: 20:56:05
Total Segments: 3

1	LRP-0	16
17	LRP-ε	29
30	LRP-γ	38

Configuration-4
Date: 2020-10-12
Time: 20:57:00
Total Segments: 3

1	LRP-γ	16
17	LRP-ε	29
30	LRP-0	38

Configuration-5
Date: 2020-10-12
Time: 20:58:28
Total Segments: 4

1	LRP-γ	9
10	LRP-custom	22
23	LRP-ε	29
30	LRP-0	38

Score	Class
11.3	plastic bag
9.9	perfume, essence
8.7	hourglass
7.8	quill, quill pen
7.6	brassiere, bra, band
7.5	red wine
7.0	beaker
6.2	wine bottle
6.1	paper towel
6.1	lab coat, laboratory

FIGURE 2.10
Model comparison view.

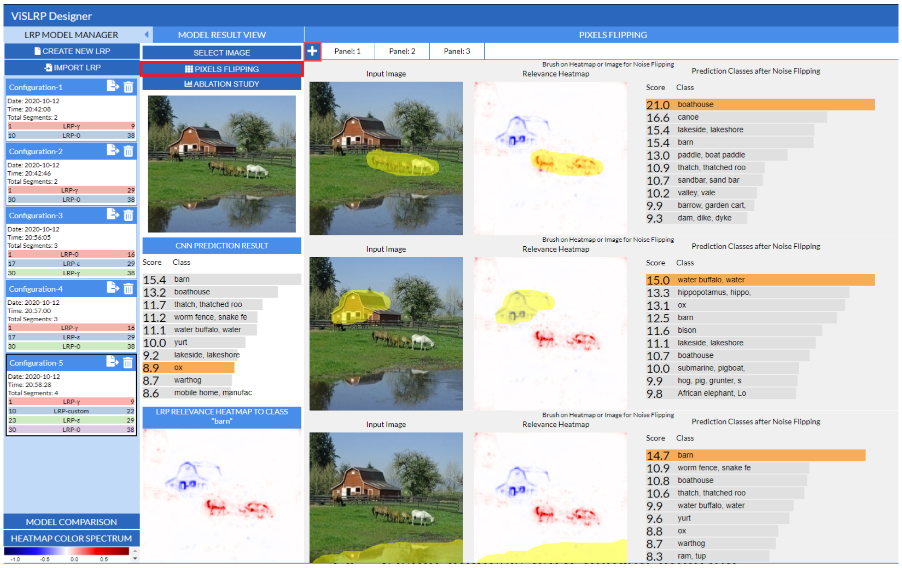
2.4 Visual Analysis with Pixel Flipping

VisLRPDesigner allows users to analyze CNN prediction by performing pixel flipping with the help of relevance information. Users can flip high- or low-relevance pixels of an input image to study the change of the CNN prediction.

A flipping interface supports users to change pixels based on the computed relevance. They can freely brush over the relevance heatmap or the input image, so that the selected pixels are flipped by multiplying the pixel value with -1. Then new CNN prediction result with the flipped input image is computed in real time.

Pixel flipping is implemented in multiple steps (Figure 2.11):

- Step 1: Click the PIXEL FLIPPING button to show the pixel flipping interface. The selected input image, its CNN prediction classes, and the LRP heatmap are shown in one row.
- Step 2: Users can brush (draw with mouse) on either the original image or the heatmap images to flip corresponding pixels.
- Step 3: The brushed pixels in the yellow area are removed and the modified image is used to regenerate CNN prediction results, which are show in a list on the right. The top class will be highlighted to be compared with the original prediction.



Configuration-1
Date: 2020-10-12
Time: 20:42:08
Total Segments: 2

1	LRP-γ	9
10	LRP-0	38

Configuration-2
Date: 2020-10-12
Time: 20:42:46
Total Segments: 2

1	LRP-γ	29
30	LRP-0	38

Configuration-3
Date: 2020-10-12
Time: 20:56:05
Total Segments: 3

1	LRP-0	16
17	LRP-ε	29
30	LRP-γ	38

Configuration-4
Date: 2020-10-12
Time: 20:57:00
Total Segments: 3

1	LRP-γ	16
17	LRP-ε	29
30	LRP-0	38

Configuration-5
Date: 2020-10-12
Time: 20:58:28
Total Segments: 4

1	LRP-γ	9
10	LRP-custom	22
23	LRP-ε	29
30	LRP-0	38

CNN PREDICTION RESULT

Score	Class
15.4	barn
13.2	boathouse
11.7	thatch, thatched roo
11.2	worm fence, snake fe
11.1	water buffalo, water
10.0	yurt
9.2	lakeside, lakeshore
8.9	ox
8.7	warthog
8.6	mobile home, manufac

LRP RELEVANCE HEATMAP TO CLASS "barn"

Prediction Classes after Noise Flipping

Score	Class
21.0	boathouse
16.6	canoe
15.4	lakeside, lakeshore
15.4	barn
13.0	paddle, boat paddle
10.9	thatch, thatched roo
10.7	sandbar, sand bar
10.2	valley, vale
9.9	barrow, garden cart,
9.3	dam, dike, dyke

Prediction Classes after Noise Flipping

Score	Class
15.0	water buffalo, water
13.3	hippopotamus, hippo,
13.1	ox
12.5	barn
11.6	bison
11.1	lakeside, lakeshore
10.7	boathouse
10.0	submarine, pigboat,
9.9	hog, pig, grunter, s
9.8	African elephant, Lo

Prediction Classes after Noise Flipping

Score	Class
14.7	barn
10.9	worm fence, snake fe
10.8	boathouse
10.6	thatch, thatched roo
9.9	water buffalo, water
9.6	yurt
8.8	ox
8.7	warthog
8.3	ram, tup

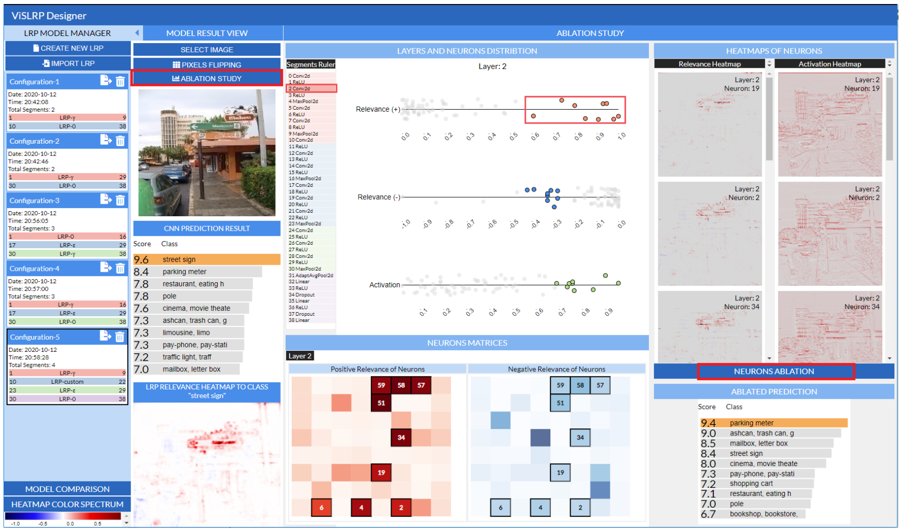
FIGURE 2.11
Relevance based pixel flipping interface.

- Step 4: Users can click the "+" button at the top left corner to add a new row. Different pixels can be brushed to compare different flipping results.

2.5 Visual Analysis with Neuron Ablation

This function allows users to perform study in the level of CNN layers and neurons. An ablation interface allows users to select CNN layers and visualize their neuron-level relevance distributions. The accumulated positive (or negative) relevance score is computed for each neuron, as well as the accumulated activation values of this neuron. These scores are visualized together with their relevance heatmaps and activation heatmaps. Based on the visual guidance, they can interactively choose individual neurons or groups of neurons for ablation study [4, 11]. The original CNN prediction is compared with the new prediction result when the selected neurons are ablated from prediction.
Neuron ablation is implemented in multiple steps (Figure 2.12):

- Step 1: Click the button of NEURONS AND ANALYSIS above the input image.
- Step 2: Select a layer of interest at the Segments Ruler. In Figure 2.12, layer-2 is selected.



Configuration-1
Date: 2020-10-12
Time: 20:42:08
Total Segments: 2

1	LRP-γ	9
10	LRP-0	38

Configuration-2
Date: 2020-10-12
Time: 20:42:46
Total Segments: 2

1	LRP-γ	29
30	LRP-0	38

Configuration-3
Date: 2020-10-12
Time: 20:56:05
Total Segments: 3

1	LRP-0	16
17	LRP-ε	29
30	LRP-γ	38

Configuration-4
Date: 2020-10-12
Time: 20:57:00
Total Segments: 3

1	LRP-γ	16
17	LRP-ε	29
30	LRP-0	38

Configuration-5
Date: 2020-10-12
Time: 20:58:28
Total Segments: 4

1	LRP-γ	9
10	LRP-custom	22
23	LRP-ε	29
30	LRP-0	38

0 Conv2d
1 ReLU
2 Conv2d
3 ReLU
4 MaxPool2d
5 Conv2d
6 ReLU
7 Conv2d
8 ReLU
9 MaxPool2d
10 Conv2d
11 ReLU
12 Conv2d
13 ReLU
14 Conv2d
15 ReLU
16 MaxPool2d
17 Conv2d
18 ReLU
19 Conv2d
20 ReLU
21 Conv2d
22 ReLU
23 MaxPool2d
24 Conv2d
25 ReLU
26 Conv2d
27 ReLU
28 Conv2d
29 ReLU
30 MaxPool2d
31 AdaptiveAvgPool2d
32 Linear
33 ReLU
34 Dropout
35 Linear
36 ReLU
37 Dropout
38 Linear

CNN PREDICTION RESULT

Score	Class
9.6	street sign
8.4	parking meter
7.8	restaurant, eating h
7.8	pole
7.6	cinema, movie theate
7.3	ashcan, trash can, g
7.3	limousine, limo
7.3	pay-phone, pay-stati
7.2	traffic light, traff
7.0	mailbox, letter box

LRP RELEVANCE HEATMAP TO CLASS "street sign"

Layer: 2
Neuron: 19

Layer: 2
Neuron: 2

Layer: 2
Neuron: 34

Positive Relevance of Neurons: 59, 58, 57, 51, 34, 19, 6, 4, 2

Negative Relevance of Neurons: 59, 58, 57, 51, 34, 19, 6, 4, 2

ABLATED PREDICTION

Score	Class
9.4	parking meter
9.0	ashcan, trash can, g
8.5	mailbox, letter box
8.4	street sign
8.0	cinema, movie theate
7.3	pay-phone, pay-stati
7.2	shopping cart
7.1	restaurant, eating h
7.0	pole
6.7	bookshop, bookstore,

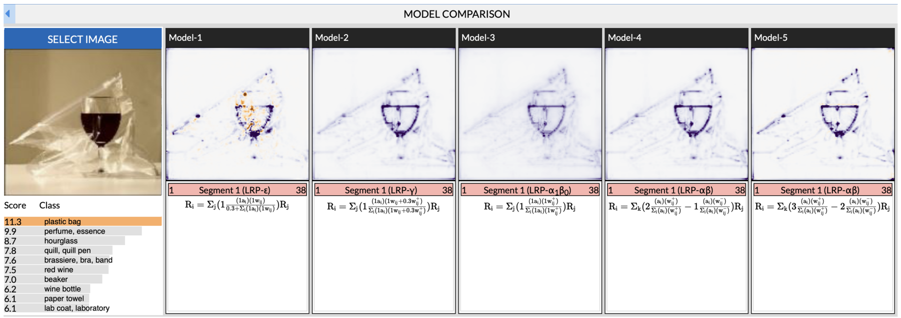
FIGURE 2.12
Relevance based neuron ablation interface.

- Step 3: In the LAYER AND NEURON DISTRIBUTION view, the neurons in the selected layer are sorted according to the positive / negative relevance summation value. They are displayed in two neuron scattering charts where each dots represent one neuron. The third neuron scattering chart shows these neurons sorted according to the activation values. Users can group and select these dots so that these neurons are ablated.
- Step 4: These neurons in the selected layer are also shown in NEURONS MATRICES view. The selected neurons from Step 3 are highlighted. Users can hover on the neuron cells to see their details. Users can also click each cell to select or deselect them from ablation.
- Step 5: The selected neurons from Steps 3-4 display their relevance heatmaps on right, together with their activation heatmaps.
- Step 6: Click the NEURONS ABLATION button, the CNN prediction is re-executed by removing the selected neurons. The new prediction results with classes and scores are shown, which can be compared with the original prediction result.

3

Example Cases

CONTENTS



Score	Class
11.3	plastic bag
9.9	perfume, essence
8.7	hourglass
7.8	quill, quill pen
7.6	brassiere, bra, band
7.5	red wine
7.0	beaker
6.2	wine bottle
6.1	paper towel
6.1	lab coat, laboratory

1 Segment 1 (LRP-ε) 38

$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{0.3+\Sigma_i(1a_i)(1w_{ij})})R_j$

1 Segment 1 (LRP-γ) 38

$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ij}^+)})R_j$

1 Segment 1 (LRP-$\alpha_1\beta_0$) 38

$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}^+)})R_j$

1 Segment 1 (LRP-αβ) 38

$R_i = \Sigma_k(2\frac{(a_i)(w_{ij}^+)}{\Sigma_i(a_i)(w_{ij}^+)} - 1\frac{(a_i)(w_{ij}^-)}{\Sigma_i(a_i)(w_{ij}^-)})R_j$

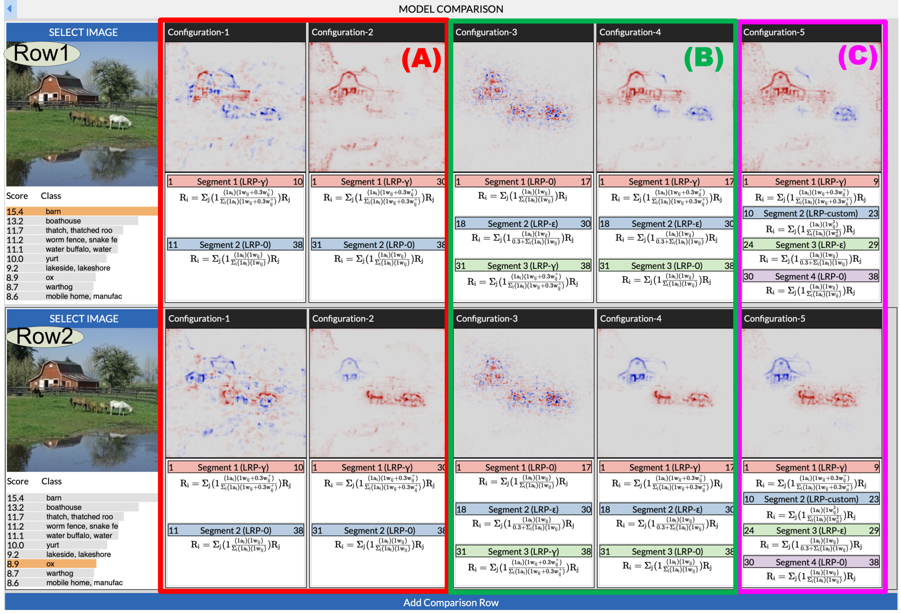
1 Segment 1 (LRP-αβ) 38

$R_i = \Sigma_k(3\frac{(a_i)(w_{ij}^+)}{\Sigma_i(a_i)(w_{ij}^+)} - 2\frac{(a_i)(w_{ij}^-)}{\Sigma_i(a_i)(w_{ij}^-)})R_j$

FIGURE 3.1
Model comparison view of popular LRP rules with an image ("A glass of wine with plastic bag"). A prediction class "plastic bag" is selected to study relevance. The relevance heatmaps of Model1 to Model5, together with their LRP rule equations, are shown for comparison. Purple/orange color refers to positive/negative relevance pixels.

3.1 Case 1: Studying Popular LRP Rules

A primary use of VisLRP is to visually explore popular LRP models. It is demanded by novice users (or students) in learning LRP concepts and gaining firsthand experience of different LRP backpropagation rules. Users can select these rules, change their parameter values, and compare them with different input images. Figure. 3.1 shows five models with the following rules (Model-1): $LRP - \epsilon$ with $\epsilon = 0.3$; (Model-2): $LRP - \gamma$ rule with $\gamma = 0.3$; (Model-3): $LRP - \alpha_1\beta_0$ with $\alpha = 1$ and $\beta = 0$; (Model-4): $LRP - \alpha_2\beta_1$ with $\alpha = 2$ and $\beta = 1$; and (Model-5): $LRP - \alpha_3\beta_2$ with $\alpha = 3$ and $\beta = 2$. Users can observe the differences of their corresponding LRP equations. In Figure. 3.1, a class "plastic bag" is selected to study the relevance on "a glass of wine and bag" image. It can be realized that Model-2 $LRP - \gamma$ can discover clearer edges of the glass and bag than the basic rule $LRP - \epsilon$ in Model-1. Model-2 removes negative contribution pixels in Model-1. In addition, $LRP - \alpha_1\beta_0$ (Model-3) adds more edge details with high relevance. Users can further compare different α and β values to understand their effects in LRP results. $LRP - \alpha_2\beta_1$ (Model-4) leads to closer result to $LRP - \gamma$ (Model-2). Comparing with Model-3, it justifies the suggested model behavior in [7]: "*if ... the heatmaps are too diffuse, replace the rule LRP-α1β0 by LRP-α2β1...*". Using $LRP - \alpha_3\beta_2$ in Model-5 further takes out more small details. In these models, the glass of wine shows big contribution to the classification of plastic bag. It indicates that sometimes the pre-defined individual LRP rules have limitation in effectively discovering salient relevance information. This can be addressed by customizing more complex LRP models.



Row1

Score	Class
15.4	barn
13.2	boathouse
11.7	thatch, thatched roo
11.2	worm fence, snake fe
11.1	water buffalo, water
10.0	yurt
9.2	lakeside, lakeshore
8.9	ox
8.7	warthog
8.6	mobile home, manufac

Configuration-1

1 Segment 1 (LRP-γ) 10
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ij}^+)})R_j$

11 Segment 2 (LRP-0) 38
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{\Sigma_i(1a_i)(1w_{ij})})R_j$

Configuration-2

1 Segment 1 (LRP-γ) 30
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ij}^+)})R_j$

31 Segment 2 (LRP-0) 38
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{\Sigma_i(1a_i)(1w_{ij})})R_j$

Configuration-3

1 Segment 1 (LRP-0) 17
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{\Sigma_i(1a_i)(1w_{ij})})R_j$

18 Segment 2 (LRP-ε) 30
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{0.3+\Sigma_i(1a_i)(1w_{ij})})R_j$

31 Segment 3 (LRP-γ) 38
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ij}^+)})R_j$

Configuration-4

1 Segment 1 (LRP-γ) 17
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ij}^+)})R_j$

18 Segment 2 (LRP-ε) 30
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{0.3+\Sigma_i(1a_i)(1w_{ij})})R_j$

31 Segment 3 (LRP-0) 38
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{\Sigma_i(1a_i)(1w_{ij})})R_j$

Configuration-5

1 Segment 1 (LRP-γ) 9
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ii}^+)})R_j$

10 Segment 2 (LRP-custom) 23
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}^2)}{\Sigma_i(1a_i)(1w_{ii}^2)})R_j$

24 Segment 3 (LRP-ε) 29
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{0.3+\Sigma_i(1a_i)(1w_{ij})})R_j$

30 Segment 4 (LRP-0) 38
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{\Sigma_i(1a_i)(1w_{ij})})R_j$

Row2

Score	Class
15.4	barn
13.2	boathouse
11.7	thatch, thatched roo
11.2	worm fence, snake fe
11.1	water buffalo, water
10.0	yurt
9.2	lakeside, lakeshore
8.9	ox
8.7	warthog
8.6	mobile home, manufac

Configuration-1

1 Segment 1 (LRP-γ) 10
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ij}^+)})R_j$

11 Segment 2 (LRP-0) 38
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{\Sigma_i(1a_i)(1w_{ij})})R_j$

Configuration-2

1 Segment 1 (LRP-γ) 30
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ij}^+)})R_j$

31 Segment 2 (LRP-0) 38
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{\Sigma_i(1a_i)(1w_{ij})})R_j$

Configuration-3

1 Segment 1 (LRP-0) 17
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{\Sigma_i(1a_i)(1w_{ij})})R_j$

18 Segment 2 (LRP-ε) 30
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{0.3+\Sigma_i(1a_i)(1w_{ij})})R_j$

31 Segment 3 (LRP-γ) 38
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ij}^+)})R_j$

Configuration-4

1 Segment 1 (LRP-γ) 17
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ij}^+)})R_j$

18 Segment 2 (LRP-ε) 30
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{0.3+\Sigma_i(1a_i)(1w_{ij})})R_j$

31 Segment 3 (LRP-0) 38
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{\Sigma_i(1a_i)(1w_{ij})})R_j$

Configuration-5

1 Segment 1 (LRP-γ) 9
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}+0.3w_{ij}^+)}{\Sigma_i(1a_i)(1w_{ij}+0.3w_{ii}^+)})R_j$

10 Segment 2 (LRP-custom) 23
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij}^2)}{\Sigma_i(1a_i)(1w_{ii}^2)})R_j$

24 Segment 3 (LRP-ε) 29
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{0.3+\Sigma_i(1a_i)(1w_{ij})})R_j$

30 Segment 4 (LRP-0) 38
$R_i = \Sigma_j(1\frac{(1a_i)(1w_{ij})}{\Sigma_i(1a_i)(1w_{ij})})R_j$

FIGURE 3.2
Model comparison view of multi-segment LRP models with an image ("barn on lake"). Row1: Relevance study for class "barn". Row2: Relevance study for class "ox". (A) Configuration 1 and 2 have two segments with different sizes; (B) Configuration 3 and 4 have three segments with the same size but different LRP rules; (C) Configuration 5 has four segments defined in Figure. 2.1.

3.2 Case 2: Exploring Customized LRP Models

LRP designers and experienced users can configure composite LRP models by applying LRP rules on multiple segments of CNN layers. In Figure. 3.2, five different configurations are shown with the "barn on lake" image. Both Row1 and Row2 use this image but perform relevance study over two different classes: "barn" and "ox", respectively. It is interesting that for this image, VGG identifies the horses as oxes, maybe because the horse bowed their heads. Please note that the operation on "ox" class actually involves horses on the image.

Figure. 3.2A shows two LRP models designed with two segments, both using $LRP - \gamma$ on Segment1 and $LRP - 0$ on Segment2. Configuration-1 defines Segment1 at layers 1-10 and Segment2 at layers 11-38. Meanwhile, Configuration-2 has Segment1 at layers 1-30 and Segment2 at layers 31-38. It can be realized that Configuration-1 has unsatisfied relevance results. Their negative and positive relevance pixels on the heatmaps do not present meaningful explanation for either "barn" or "ox" classes. In contrast, Configuration-

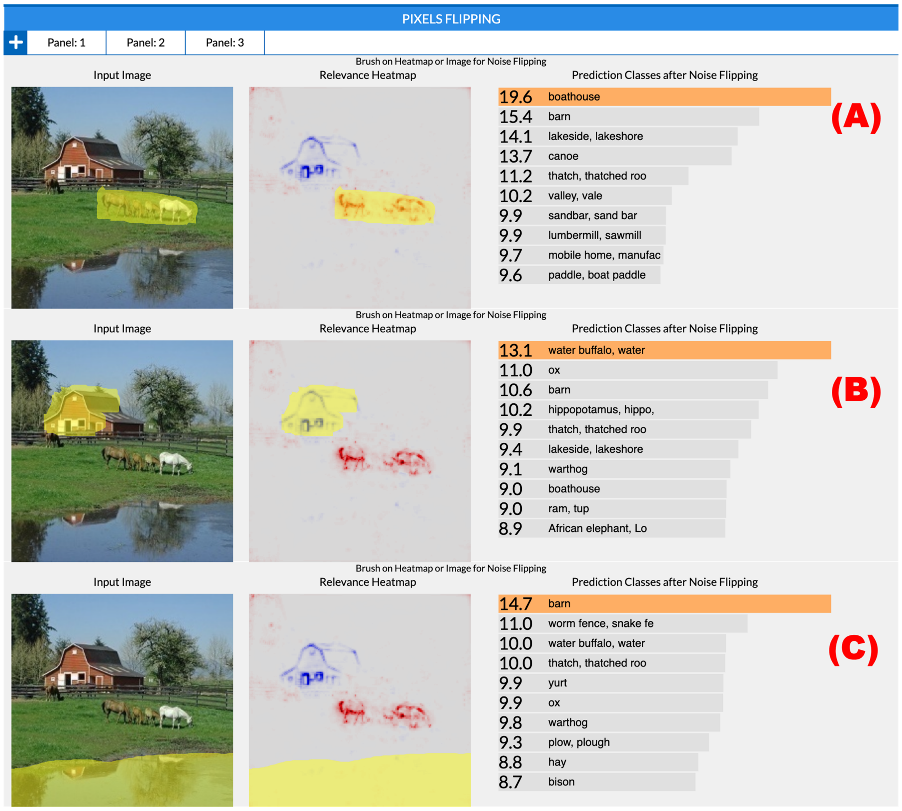
2 performs very well. In Row1, the barn house is discovered while the horses are not emphasized in the relevance heatmap. In Row2, the horses (i.e., ox class) are identified with high positive relevance, and the barn is realized with negative contributions. This example shows that different segment sizes can lead to very different LRP behaviors.

In Figure. 3.2B, Configuration-3 and Configuration-4 are defined on three fixed segments. In Configuration-3, $LRP{-}0$, $LRP{-}\epsilon$, and $LRP{-}\gamma$ are applied on Segment1, 2 and 3, respectively. In Configuration-4, $LRP - \gamma$, $LRP - \epsilon$, and $LRP - 0$ are used on Segment1, 2 and 3, respectively. Configuration-3 fails to achieve meaningful LRP results. On the other hand, Configuration-4 gives very good relevance heatmaps. In Row1, it identifies the barn house as positive contributor and the horses as negative contributor. In Row2, it discovers the horses with positive relevance and the barn house with negative relevance. This case shows that different rules on the same segments can also generate very different LRP results.

In Figure. 3.2C, four segments are defined in Configuration-5 which uses $LRP - \gamma$, $LRP - \epsilon$, and $LRP - 0$ in Segment1, Segment3, and Segment4 (similar to Segment 1-3 in Configuration-4). A custom LRP rule is designed and inserted as Segment2. The parameter settings of this configuration are shown in Figure. 2.1. In comparison to Configuration-4, this model detects more positive pixels. As shown in Row2, more pixels of the two small horses (in the middle of two big horses) are linked to the “ox” class. But it does not show big difference in Row1. This example shows the exploratory process of LRP model design.

3.3 Case 3: Relevance Based Pixel Flipping

VisLRP allows users to analyze CNN prediction by performing pixel flipping with the help of relevance information. In Figure. 3.3, users apply three different pixel flipping operations on different parts of the “barn on lake” image. Here the relevance heatmap of “ox” class from Configuration-4 of Figure. 3.2 is selected. In Figure. 3.3A, users brush on the relevance heatmap to remove positive contributor pixels. The new CNN prediction result shows the top class as “boathouse”, which reflects the effect of CNN prediction after removing horses. In Figure. 3.3B, the barn house with negative relevance is removed. The new prediction shows top classes “water buffalo” and “ox”, which helps understand the prediction behavior. Finally, in Figure. 3.3C, users directly brush on the input image to remove the pond. The new result shows “barn” and “worm fence” as top classes. Here, “boat house”, which is the second class in the original classification (see Figure. 2.1), is no longer discovered. It indicates how the water surface contributes to the classification.


PIXELS FLIPPING
+
Panel: 1
Panel: 2
Panel: 3
Brush on Heatmap or Image for Noise Flipping
Input Image
Relevance Heatmap
Prediction Classes after Noise Flipping
(A)
(B)
(C)

(A)

19.6	boathouse
15.4	barn
14.1	lakeside, lakeshore
13.7	canoe
11.2	thatch, thatched roo
10.2	valley, vale
9.9	sandbar, sand bar
9.9	lumbermill, sawmill
9.7	mobile home, manufac
9.6	paddle, boat paddle

(B)

13.1	water buffalo, water
11.0	ox
10.6	barn
10.2	hippopotamus, hippo,
9.9	thatch, thatched roo
9.4	lakeside, lakeshore
9.1	warthog
9.0	boathouse
9.0	ram, tup
8.9	African elephant, Lo

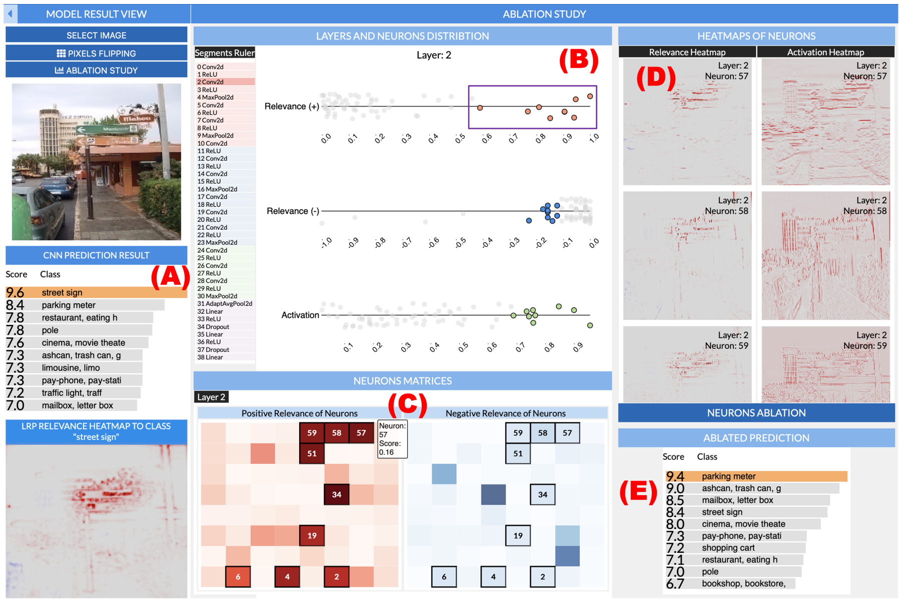
(C)

14.7	barn
11.0	worm fence, snake fe
10.0	water buffalo, water
10.0	thatch, thatched roo
9.9	yurt
9.9	ox
9.8	warthog
9.3	plow, plough
8.8	hay
8.7	bison

FIGURE 3.3
Relevance-based pixel flipping study with image "barn on lake". Users brush on the image or the relevance heatmap to flip pixels, so as to study new CNN prediction results. Multiple rows are added with different flipping operations for comparison study.

3.4 Case 4: Relevance Based Neuron Ablation

By selecting a "street view" image, as shown in Figure. 3.4A, users find the prediction result with top three classes as "street sign", "parking meter", and "restaurant". By selecting class "street sign", the relevance heatmap shows high relevance pixels of signs to this class. For ablation study, users can click on the ruler to select the convolution layer 2. The distribution charts show the neuron points in this layer. Users select a group of neuron points with high positive relevance scores, as shown inside the purple box in Figure. 3.4B. It can be seen these neurons also have large activations (green points) and small negative relevance values (blue points). Users further explore these neurons in the matrices of layer 2 in Figure. 3.4C. By observing Figure. 3.4D, it can be



CNN PREDICTION RESULT

Score	Class
9.6	street sign
8.4	parking meter
7.8	restaurant, eating h
7.8	pole
7.6	cinema, movie theate
7.3	ashcan, trash can, g
7.3	limousine, limo
7.3	pay-phone, pay-stati
7.2	traffic light, traff
7.0	mailbox, letter box

ABLATED PREDICTION

Score	Class
9.4	parking meter
9.0	ashcan, trash can, g
8.5	mailbox, letter box
8.4	street sign
8.0	cinema, movie theate
7.3	pay-phone, pay-stati
7.2	shopping cart
7.1	restaurant, eating h
7.0	pole
6.7	bookshop, bookstore,

FIGURE 3.4

Relevance-based neuron ablation study with an image ("street view"). (A) Relevance result view for a top class "street sign"; (B) Neuron points from a selected layer 2 showing their positive relevance, negative relevance, and activation distributions. Users select a group of neurons (in a purple rectangle); (C) Neuron matrices for detail study of positive/negative relevance; (D) Relevance heatmaps and activation heatmaps of selected neurons; (E) Prediction results after ablation with a top class "parking meter".

seen that Neuron 57, 58, 59 (more to be observed by scrolling down further) show different neuron activations but their high-relevance pixels are mostly on the street signs. Figure. 3.4E shows the new prediction result after these neurons in layer 2 are removed from the CNN computation. Now "street sign" is not the top class while "parking meter" becomes the top class. So users can partly explain the behaviors of these neurons.

References

[1] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10(7), 2015.

[2] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *MM 2014 - Proceedings of the 2014 ACM Conference on Multimedia*, pages 675–678, 2014.

[3] Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus Robert Müller. Unmasking Clever Hans predictors and assessing what machines really learn. *Nature Communications*, 10(1), 2019.

[4] Peter E. Lillian, Richard Meyes, and Tobias Meisen. Ablation of a Robot's Brain: Neural Networks Under a Knife. 2018.

[5] Grégoire Montavon, Alexander Binder, Sebastian Lapuschkin, Wojciech Samek, and Klaus Robert Müller. Layer-Wise Relevance Propagation: An Overview. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11700 LNCS:193–209, 2019.

[6] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus Robert Müller. Explaining nonlinear classification decisions with deep Taylor decomposition. *Pattern Recognition*, 65:211–222, 2017.

[7] Grégoire Montavon, Wojciech Samek, and Klaus Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing: A Review Journal*, 73:1–15, 2018.

[8] Wojciech Samek, Alexander Binder, Sebastian Lapuschkin, and Klaus Robert Muller. Understanding and Comparing Deep Neural Networks for Age and Gender Classification. *Proceedings - 2017 IEEE International Conference on Computer Vision Workshops, ICCVW 2017*, 2018-Janua:1629–1638, 2017.

[9] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015.

[10] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 07-12-June:1–9, 2015.

[11] Y. Vishnusai, Tejas R. Kulakarni, and K. Sowmya Nag. Ablation of Artificial Neural Networks. pages 453–460, 2020.