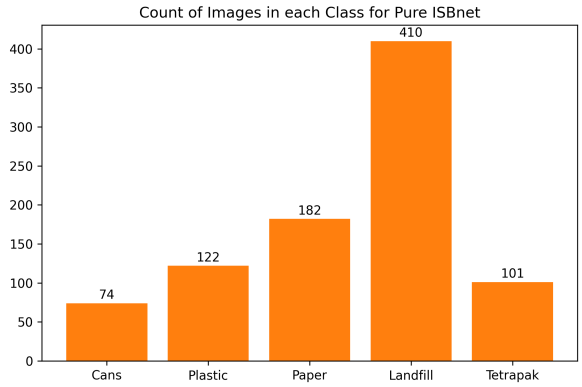

Fig. 2. Class Distribution of ISBNet

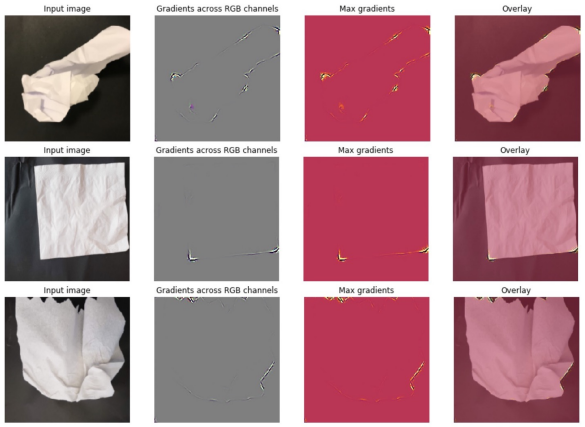

Fig. 3. Saliency maps from a baseline ResNet50 classifier. The images from top to bottom: crumpled piece of paper (paper), a napkin (landfill), cumpled piece of tissue (landfill).

Figure 2. The data acquisition process involved using a piece of black poster paper as a background; this would create enough contrast for trash belonging to the paper category. These pictures were taken with an iPhone 8 and an iPhone XS. We recorded the trash can in which the piece of trash originated from and any trash-generating landmarks nearby. Section IV details the encoding and formatting of these meta labels. Data augmentation techniques were performed on the images due to the limited size of each class. This included grey-scaling, random rotation, re-scaling, and shearing. Mean subtraction and normalization were also performed on the dataset.

Importantly though, it is not feasible to collect the exact time in which the piece of trash was disposed of. Trash that can be realistically imaged is retrieved from bins where it was disposed of at an unknown time. Regular trash cans also do not possess sensors that can detect trash being thrown into the bin and simultaneously record the time and associated image. Without sensors, the only workaround is to station a person at a trash can who is collecting the trash and recording its information in real time, which is simply not achievable in the vast majority of cases. As a result of this inherent shortcoming that exists when procuring a trash dataset possessing metadata, we approximated time distributions for trash cans that could be learned by the model. This methodology and its underlying theory are discussed in the *Time* subsection within the *Metadata* section.

IV. Metadata

Metadata of all kinds can be collected through sensors in a smart trash can where our model could be implemented, such as location of the trash can and its distance to landmarks, or time of day. Incorporating metadata as extra inputs to an image-based neural network decreases the likelihood for *feature confusion*. Items of trash belonging to different categories may have similar features. A network that solely depends on image features is often not able to differentiate between these objects. This is exemplified in Figure 3.

Saliency maps [21] of an image-based trash classifier were generated for images of paper and tissues/napkins, which belong to the landfill class. These are two of the classes with lowest precision for image-based classifiers. The trained classifier shown in Figure 3 incorrectly predicted both landfill pictures, the napkin and tissue, as belonging to the paper class. The saliency maps illustrated that the image-based model falsely associated rigid edges and crumples in the tissue and napkin with the paper class. Exposing the network to additional time and location information will increase its ability to discern images of similar features, decreasing the likelihood for feature confusion.

We used two fields of metadata in our network: location and distance, as well as time. The methodology implemented to transform these fields into inputs for our network is outlined in the following subsections: *Location and Distance* and *Time*.

A. Location and Distance

The geographical location of a trash can allows us to identify its proximity to certain landmarks. We define a *landmark*, in the context of trash classification, as an identifiable area that skews the distribution of the type of trash and amount of trash found in a proximal trash can. Landmarks may affect trash generation either through the inherent nature of these landmarks, or through the increased foot traffic experienced by these areas. Examples of landmarks that we identified are cafeterias (eating may produce more contaminated and food-related trash), printers (recyclable paper would be more common next to a printer), or entrances/exits (the large flow of people means more trash is likely to be deposited in the nearby bins).

We acquired detailed, scaled blueprints, with trash can locations marked, of the two floors where ISBNet was collected. The limited number of trash cans meant trash belonging to different landmarks was grouped together. Figure 4 showcases trash can 8A, an example of this, which received a large amount of trash due to its proximity to a variety of significant landmarks: the cafeteria, library, a lounge, a stairwell, a printer, and bathroom. In turn, a substantial quantity of trash generated from those landmarks was

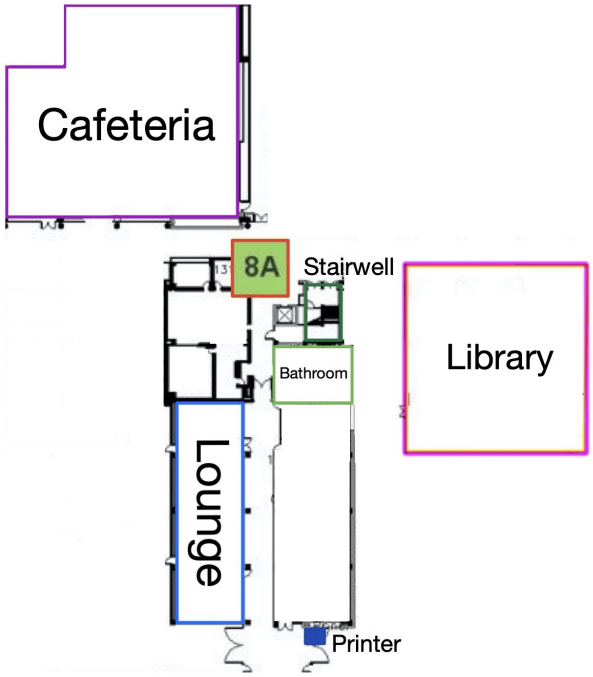

Fig. 4. Trash can "8A" with a cafeteria, stairwell, bathroom, printer, lounge, and library as proximal landmarks.

present in 8A, and thus assigned the corresponding trash can metadata. This introduced a larger degree of homogeneity to the metadata than preferred. However, this issue is alleviated by introducing landmark distances, which is introduced later in this subsection.

Using these blueprints, we identified 11 types of trash-generating landmarks around the school. These landmarks are listed below.

- Entrance/Exit
- Bathroom
- Lounge
- Stairwell
- Cafeteria
- Theater
- Gym
- Pool
- Printer
- Couch Area
- Library

A trash can's *radius of proximity* is determined as the average of the distances between the trash can and its nearest neighboring bins. Thus, a trash can's *proximal landmarks* are defined as the landmarks in the set of all landmarks listed above that are within its radius of proximity.

In order to input the location and distance information into the network, a few transformations were conducted. Each trash can was annotated with a binary vector, $\overrightarrow{v}_{prox} \in \{0,1\}^{11}$, that represents its proximal landmarks. Landmarks that exceeded its radius of proximity around the trash can were indicated by a 0 entry. Another vector $\overrightarrow{v_{\mathrm{d}}} \in \mathbb{R}^{11}$ describes the diagonal distance between the center of the trash can and each landmark. Element-wise multiplication between these two vectors was performed so only distances from proximal landmarks are considered. This process is shown below:

$$\overrightarrow{v_{\mathrm{c}}} = \overrightarrow{v_{\mathrm{d}}} \odot \overrightarrow{v}_{prox}$$

The non-zero entries of this new vector, $\overrightarrow{v_{\mathrm{c}}}$, represent the meter distances between the trash can and its proximal landmarks. The vector $\overrightarrow{v_{\mathrm{c}}}$ is normalized into a unit vector using the L^2 norm. This converts the absolute meter distances into relative distances. This unit vector of $\overrightarrow{v_{\mathrm{c}}}$ is expressed as $\hat{\boldsymbol{v}}_{\mathbf{c}}$, where the non-zero relative distances are inversely proportional to their weight.

However, the zero entries in this vector do not represent relative distance, rather they represent the absence of a proximal landmark. Theoretically, these zero-values describe landmarks that are an infinite distance away. However, for numerical computation, we chose a value β that is sufficiently large to represent this behavior. This β value replaces the zero-values in the unit vector $\hat{\boldsymbol{v}}_{\mathbf{c}}$.

A component-wise negative logarithmic transformation, using the natural logarithm, was applied to this unit vector $\hat{\boldsymbol{v}}_{\mathbf{c}}$. This transformation is reflective of the tendency of consumers to throw away trash in the trash can closest to the trash-generating landmark. Therefore, the relative influence of a landmark decays exponentially as the distance between the landmark and the trash can increases linearly. This final location vector is concatenated with a time vector representing the time metadata, which is described in the next section.

B. Time

Time of day refers to the time when a piece of trash was thrown into the bin. During noon, we would expect that the large amount of trash generated is associated with lunch time. More likely than not, these pieces of trash would belong to the landfill or tetra pak categories. A piece of trash disposed during noon is highly likely to be associated with the cafeteria. Thus, the significance of a trash can's proximal landmarks is associated with time.

Approximating the time distribution for specific trash cans through observing foot traffic is perferable over inputting a real-time, time, expressed as a scalar. The limited size of the dataset decreases the probability that the collected data will encompass the probability distribution for each bin. Moreover, although collecting this data during production-time is intuitive, it is unfeasible to acquire this information without existing sensors and infrastructure in place.

An alternative and more generalizable method, described in this section, to account for this a priori probability distribution is needed. By identifying multiple peaks for each bin the model will be able to learn the correlation between each of these peaks and its respective proximal landmarks.

Thus, during production time, the arrival time expressed as a one-hot vector will contain a singular peak. This is

then used by the pre-trained model to discriminate between relatively important and unimportant proximal landmarks.

Each landmark is associated with a foot traffic distribution. This foot traffic distribution varies between landmarks. We categorized all foot traffic distributions into three categories: multi-model, normal, and uniform.

Landmarks characterized as multi-modal demonstrate increased foot traffic during multiple, regular, scheduled times of day. Consider the following example of the school cafeteria, which can be described using a multi-modal foot traffic distribution:

The cafeteria is mostly accessed during lunch periods and directly after school. These time periods are 12pm and 4pm respectively. Thus, the foot traffic distribution for the cafeteria is modelled through the composed of two truncated normal distributions. A truncated normal distribution was used as we assumed the cafeteria experiences no traffic before it opens and after it closes, 9am and 6pm respetively. The means of the two component foot traffic distributions are 12pm and 4pm, while the variances were estimated by surveying the traffic around the cafetera. Let the random variables X_1 and X_2 describe the two truncated normal distributions that compose the foot traffic distribution of the cafeteria. These random variables are initialized such that:

$$X_1 \sim N(\mu = 12\text{pm}, \sigma^2 = (15\text{min})^2, a = 9\text{am}, b = 6\text{pm})$$
$$X_2 \sim N(\mu = 4\text{pm}, \sigma^2 = (15\text{min})^2, a = 9\text{am}, b = 6\text{pm})$$

where X conditional on $a < X < b$ has a truncated normal distribution.

The probability density functions (PDF) of these two truncated normal distributions are defined as:

$$p_1(x) = \begin{cases} \frac{1}{15\text{min}} \frac{\phi(\frac{x-12\text{pm}}{15\text{min}})}{\Phi(\frac{6\text{pm}-12\text{pm}}{15\text{min}}) - \Phi(\frac{9\text{am}-12\text{pm}}{15\text{min}})} & 9\text{am} < x < 6\text{pm}, \\ 0 & \text{otherwise.} \end{cases}$$

$$p_2(x) = \begin{cases} \frac{1}{15\text{min}} \frac{\phi(\frac{x-4\text{pm}}{15\text{min}})}{\Phi(\frac{6\text{pm}-4\text{pm}}{15\text{min}}) - \Phi(\frac{9\text{am}-4\text{pm}}{15\text{min}})} & 9\text{am} < x < 6\text{pm}, \\ 0 & \text{otherwise.} \end{cases}$$

Here, $\phi(\cdot)$ is the probability density function of the standard normal distribution, while $\Phi(\cdot)$ is its cummulative distribution function.

The composite foot traffic distribution, $p_{\text{cafeteria}}$, of the cafeteria is determined by averaging the two PDFs, p_1 and p_2. This is shown below:

$$p_{\text{cafeteria}}(x) = \frac{p_1(x) + p_2(x)}{2}$$

An average was to combine these distributions to preserve the peaks of the respective truncated normal distributions, while maintaining constant area under the PDF.

Landmarks characterized as normal, experience activity clustered around a central mean, adhering to a single truncated normal distribution. Landmarks that follow such a distribution include the theater which, on regular days, only experiences traffic after lunch time for assemblies.

Landmarks that are described by uniform distributions are accessed throughout the day with similar levels of activity, such as bathrooms and printers. Consider the following example of a printer: We assume that the foot traffic for all printers experience a uniform probability distribution during the 8am to 6pm school day. Let random variable X_{printer} describe the printer's uniform distribution. This random variable is initialized such that:

$$X_{\text{printer}} \sim \text{Uni}(t_i = 8\text{am}, t_f = 6\text{pm})$$

The probability distribution of this uniform distribution is shown below:

$$p_{\text{printer}}(x) = \begin{cases} \frac{1}{6\text{pm}-8\text{am}} & 8\text{am} < x < 6\text{pm}, \\ 0 & \text{otherwise.} \end{cases}$$

With the foot traffic distributions for each landmark characterized and determined, this needed to be transformed into a viable input representative of the bin.

First, each bin was assigned a convolutional probability distribution, which is the average of all foot-traffic probability distributions from the bin's proximal landmarks. For a given bin b_{m} which has n proximal landmarks, its corresponding probability density function (PDF) is defined as $f_{\text{m}}(x)$. b_{m}'s probability density function with respect to time can be described using the following:

$$f_{\text{m}}(x) = \sum_{i}^{n} \frac{1}{n} f_{\text{i}}(x)$$

where $f_i(\cdot)$ is the truncated probability density function of foot traffic of the ith proximal landmark.

This convolutional PDF, which models the trash-generating distribution for bin b_{m}, is discretized by sampling over one-hour interval in the sample space $[8\text{am}, 6\text{pm}]$. These discretized values form a new vector $\vec{v}_{\text{t}}$, where component i of $\vec{v}_{\text{t}}$ is indexed as:

$$\vec{v}_{\text{t}}[i]$$

where $\vec{v}_{\text{t}}[0]$ is the first componet in $\vec{v}_{\text{t}}$ and so forth.

The components of this vector are initialized by determining the area under this newly defined trash can convolutional PDF, f_{m}. This process is described below:

$$\vec{v}_{\text{t}}[0] = \int_{8\text{am}}^{9\text{am}} f_{\text{m}}(x)dx$$
$$\vec{v}_{\text{t}}[1] = \int_{9\text{am}}^{10\text{am}} f_{\text{m}}(x)dx$$
$$\vdots$$
$$\vec{v}_{\text{t}}[10] = \int_{5\text{pm}}^{6\text{pm}} f_{\text{m}}(x)dx$$

This creates a vector of length 10, where each component corresponds to the probability that a piece of trash is thrown in a specific one-hour interval during an average school day (8am - 6pm).

The vector $\vec{v}_{\text{t}}$ is then normalized into the unit vector using the L^2 norm. The unit vector of $\vec{v}_{\text{t}}$ is denoted by $\hat{\boldsymbol{v}}_{\text{t}}$.

The time metadata $\hat{\boldsymbol{v}}_{\mathbf{t}}$ and location metadata $\hat{\boldsymbol{v}}_{\mathbf{c}}$ are concatenated such that:

$$\sum(\hat{\boldsymbol{v}}_{\mathbf{t}}, \hat{\boldsymbol{v}}_{\mathbf{c}}) = (\hat{\boldsymbol{v}}_{\mathbf{c}}[0], \ldots, \hat{\boldsymbol{v}}_{\mathbf{c}}[11], \hat{\boldsymbol{v}}_{\mathbf{t}}[0], \ldots, \hat{\boldsymbol{v}}_{\mathbf{t}}[10])$$

This vector, of length 21, forms the metadata input for ThanosNet.

The time metadata $\hat{\boldsymbol{v}}_{\mathbf{t}}$ and location metadata $\hat{\boldsymbol{v}}_{\mathbf{c}}$ are concatenated such that:

$$\sum(\hat{\boldsymbol{v}}_{\mathbf{t}}, \hat{\boldsymbol{v}}_{\mathbf{c}}) = (\hat{\boldsymbol{v}}_{\mathbf{c}}[0], \ldots, \hat{\boldsymbol{v}}_{\mathbf{c}}[10], \hat{\boldsymbol{v}}_{\mathbf{t}}[0], \ldots, \hat{\boldsymbol{v}}_{\mathbf{t}}[11])$$

This vector, of length 21, forms the metadata input for ThanosNet.

V. Experiments

This section is organized into four subsections. The first section, *Weighted Loss Function*, explains and justifies the loss function we employed. In the second section *Baseline Experiments*, transfer learning methods are used with pre-trained ImageNet models to establish a baseline score for comparison against ThanosNet. Later, *Metadata Experiments* goes into detail regarding the architecture and performance of our ThanosNet model. The last section, *Results*, analyzes the performance ThanosNet relative to the baseline models established in the *Baseline Experiments* subsection.

For all of our experiments, we utilized stratified cross-validation training with a fold count of 5, each fold being trained over 50 epochs. The performance of each model was evaluated through average maximum validation macro F_1 in each fold, and the corresponding average validation loss in the same epoch. As the dataset was highly imbalanced, we used macro F_1 as the validation metric to gauge the precision and recall of our model, which is defined by the following,

$$F_1 = 2 \cdot \frac{pr \cdot re}{pr + re}$$

where pr and re represent precision and recall, respectively.

A. Weighted Loss Function

For this multi-class classification model, let X denote the input image and metadata. The scalar $y_{\text{label}} \in \{0, 1, 2, 3, 4\}$ denotes the label representing the class that input X belongs to: cans, landfill, paper, plastic, and tetra pak respectively. $\overrightarrow{y_{\text{predict}}} \in \mathbb{R}^5$ represent the models prediction of the input X. Component i of the prediction vector $\overrightarrow{y_{\text{predict}}}$ is indexed by $\overrightarrow{y_{\text{predict}}}[i]$. The standard cross-entropy loss function is defined as the following:

$$L(\overrightarrow{y_{\text{predict}}}, y_{\text{label}}) = -\overrightarrow{y_{\text{predict}}}[y_{\text{label}}] + log\left(\sum_j e^{\overrightarrow{y_{\text{predict}}}[j]}\right)$$

The trained model defined by the standard cross-entropy loss function above shows poor performance as it demonstrates a poor recall rate. The reason stems from the fact that ISBNet is imbalanced. Therefore, a weighted loss function was used in place of the standard cross-entropy loss, where the weights of a class are inversely proportional its class size. The weight of class i is defined by

$$\omega_i = \frac{\sum_j^n \|j\|}{\|i\|}$$

where n denotes the set of all classes: cans, landfill, paper, plastic, and tetra pak. $\|j\|$ is the size of class j, and $\|i\|$ is the size of class i.

Thus, the weighted cross-entropy loss function is defined by the following:

$$L(\overrightarrow{y_{\text{predict}}}, y_{\text{label}}) = \omega_{y_{\text{label}}}\left(log\left(\sum_j e^{\overrightarrow{y_{\text{predict}}}[j]}\right) - \overrightarrow{y_{\text{predict}}}[y_{\text{label}}]\right)$$

In our experiments, we discovered that using such a weighted loss function resulted in better performance.

B. Baseline Experiments

We experimented with VGG16, ResNet50 [22], and DenseNet169 as feature extractors for our image-based, baseline models. These are networks that have performed well in prior literature [19], [13], [17], [20] and thus are a valuable benchmark for comparison. The pre-trained ImageNet model had its respective classification layers removed and replaced with three fully connected layers with ReLU activation functions in between. During training, regularization techniques including batch normalization, dropout, and l_2 regularization were applied. We trained with a batch size of 32. The Adam optimizer was used with a learning rate of 10^{-5} and weight decay of 10^{-10}.

All image inputs were first resized to 256 $\times$ 256, then center-cropped to 224 $\times$ 224 to match the ImageNet feature extractor input parameters. These images were then normalized based on the mean and standard deviation in the ImageNet training set: $\mu_r = 0.485, \mu_g = 0.456, \mu_b = 0.406$ and $\sigma_r^2 = 0.229, \sigma_g^2 = 0.224, \sigma_b^2 = 0.225$. A random-resized crop was experimented with during training instead of the center crop. This resulted in an extremely volatile training F_1 score, preventing the model from converging. This suggests that object localization plays an important factor in classification as there was a large variance in trash size. The random resized crop could be cropping out important features.

C. Metadata Experiments

To incorporate metadata into the network, two attachments to the existing baseline networks were proposed: a bilinear [23] attachment, and an additive attachment. The architecture of these two variants of ThanosNet, AdditiveThanosNet and BilinearThanosNet, are shown in Figure 5 and Figure 6, respectively.

Each of these two variants seek to replicate an intuition behind small-scale trash classification. Metadata, such as the location and time distribution of the trash can, provides information that inherently skews the prediction of a trash item. Therefore, the first variant, AdditiveThanosNet, replicates this inherent bias by using the additive attachment as a bias parameter for the prediction layer.