Gurdarshan Singh, 2026, 14:4
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

A Fine-Tuned Vision Transformer with Grad-CAM Explainability for Real-Time 36-Class Waste Classification and REST API Deployment

Gurdarshan Singh, Shivam, Yash Singh
Panipat Institute of Engineering and Technology Department of Computer Science and Engineering

Abstract- Sorting waste by hand is slow, error-prone, and still one of the biggest bottlenecks in recycling streams. This paper presents WasteVision AI, an end-to-end system for fine-grained waste classification built around a Vision Transformer (ViT-Base/16), fine-tuned on the 36-class TriCascade WasteImage dataset (35,264 images) through a two-stage transfer-learning schedule. The backbone is first warmed up as a whole with a frozen classifier head under a class-weighted, label-smoothed cross-entropy loss, after which the last four transformer blocks are unfrozen and fine-tuning continues. On a held-out test set of 3,527 images, the resulting model reaches 95.66% top-1 accuracy, 98.27% top-3 accuracy, and a macro-averaged F1-score of 0.93. To explain its predictions, we adapt Grad-CAM to work on the transformer's patch-token hidden states, producing pixel-wise heatmaps that localise the discriminative regions of each waste item; predictions below 50% confidence are flagged "Unknown Waste" rather than forced into a class. The whole pipeline is exposed through a lightweight Flask REST API that returns the predicted class, a calibrated confidence score, and a base64-encoded Grad-CAM overlay in a single call. Against a recent three-stage DP-CNN/Ensemble-ELM benchmark evaluated on the same dataset, our single-stage ViT pipeline gains over 10 percentage points of fine-grained (36-class) accuracy, though at a higher parameter cost — a trade-off worth weighing when choosing between transformer- and CNN-based deployments for waste sorting.

Keywords— Waste classification, Vision Transformer, transfer learning, Grad-CAM, explainable AI, REST API, sustainable recycling.

I. INTRODUCTION

Global municipal solid waste generation is set to rise sharply in the coming decades, and poor sorting at the source remains one of the biggest obstacles to effective recycling and resource recovery [1]. Automated, vision-based classification offers a practical way to complement manual sorting, cutting labour costs and speeding up throughput in recycling facilities. Convolutional architectures — from heavyweight transfer-learning backbones to purpose-built lightweight networks — have already been shown to classify waste images with high accuracy, but two gaps stand out in the applied literature. Most systems are still evaluated offline, with little discussion of how the trained model would actually be packaged for real-world inference, and relatively few studies push transformer-based architectures, despite their strong transfer-learning track record in general image recognition, into fine-grained (>30 class) waste taxonomies.

This paper sets out to close both gaps. We fine-tune a Vision Transformer (ViT-Base/16), pretrained on ImageNet-21k, on the 36-class TriCascade WasteImage dataset using a two-stage procedure, evaluate it with standard classification metrics alongside a Grad-CAM-based explainability layer adapted to the transformer's patch-token representation, and package it as a Flask REST API that a front-end or embedded client can call directly. Our contributions are as follows.

A two-stage ViT fine-tuning protocol — head warm-up followed by partial backbone unfreezing under a class-weighted, label-smoothed loss — that reaches 95.66% top-1 accuracy on a 36-class, 3,527-image test split, outperforming a recent lightweight CNN-ensemble baseline on the same source dataset.

A gradient-based explanation method for ViT waste classifiers that works directly on patch-token hidden states, reshaping the resulting relevance vector into a 14×14 spatial map that can be overlaid on the input image.

A minimal, reproducible deployment path — a Flask/Flask-CORS inference service that loads the fine-tuned checkpoint, runs prediction and Grad-CAM generation in a single request, and applies a confidence-based rejection rule — along with an automated integration-test script.

An ablation study that quantifies how much accuracy partial backbone unfreezing adds over head-only linear probing, plus a mathematical formalisation of the model's self-attention, class-weighted label-smoothed loss, patch-token Grad-CAM, and confidence-scoring components.

The rest of the paper is organised as follows. Section II reviews related work; Section III covers the dataset, model, training procedure, explainability method, and deployment pipeline; Section IV reports results; Section V discusses limitations and future work; and Section VI concludes.

II. RELATED WORK

1. Transfer-Learning-Based Waste Classification

Early deep-learning approaches to waste classification leaned heavily on transfer learning from ImageNet-pretrained CNNs. Zhang et al. [5] fine-tuned DenseNet169 on NWNU-TRASH (2,528 images, 5 classes) and reported accuracy above 82%, while Vo et al. [6] proposed a ResNeXt-based DNN-TC model that reached 98% and 94% accuracy on VN-trash and TrashNet, respectively. Neelakandan et al. [7] paired YOLOv5 detection with a stacked sparse autoencoder for industrial waste, achieving over 99% accuracy on a 6-class, 2,467-image benchmark.

2. Lightweight and Multi-Stage Models

Later work turned to lightweight architectures to cut the cost of large transfer-learning backbones. Zheng et al. [8] introduced Focus-RCNet with knowledge distillation, reaching 92% accuracy on TrashNet with only 0.53M parameters; Jin et al. [9] deployed an improved MobileNetV2 for real-time sorting hardware; and Cheema et al. [10] paired a VGG16 classifier with an IoT sensing pipeline. Closest to the present study, Nahiduzzaman et al. [1] introduced the TriCascade WasteImage dataset — the same source collection used here — along with a parallel depth-wise separable CNN (DP-CNN) feature extractor and an ensemble extreme-learning-machine (En-ELM) classifier. Their staged accuracy came to 96.0% (2 classes), 91.0% (9 classes), and 85.25% (36 classes) with a compact 1.09M-parameter model, backed by SHAP/Grad-CAM explainability and a servo-controlled sorting prototype.

3. Vision Transformers and Explainability

Once pretrained on large corpora and fine-tuned downstream, Vision Transformers [2] match or exceed CNN performance, yet their use in fine-grained waste sorting — and the patch-level explainability that comes with it — remains comparatively underexplored. Grad-CAM [3] was designed for convolutional feature maps and needs adapting before it can operate on a transformer's sequence of patch-token hidden states; we develop that adaptation in Section III-D.

III. MATERIALS AND METHODS

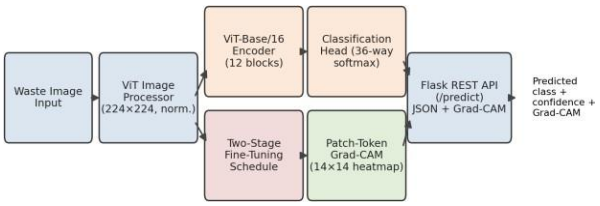

Fig. 1. End-to-end system architecture: preprocessing, ViT backbone, two-stage fine-tuning, patch-token Grad-CAM, and Flask REST API.

1. Dataset and Preprocessing

The model is trained on the TriCascade WasteImage dataset — 35,264 images across 36 fine-grained categories such as A_Foods, C_Cardboard, K_Beverage Cans, T_Clothes, and Z_G_Battery. We parsed the class directories into an image-path/label index and split it, in stratified fashion, into 28,211 training, 3,526 validation, and 3,527 test images so

that class proportions were preserved across splits. Training images were augmented on the fly with horizontal flip (p=0.5), vertical flip (p=0.15), and random rotation (±20°), and every image was processed with the Hugging Face ViTImageProcessor matched to the pretrained checkpoint — resized to 224×224, rescaled to [0,1], and normalised channel-wise with a mean/std of 0.5.

2. Model Architecture

The backbone is google/vit-base-patch16-224-in21k, pretrained on ImageNet-21k, with its head swapped out for 36 output classes. The 224×224 input is split into 196 non-overlapping 16×16 patches plus a learnable [CLS] token and passed through the standard multi-head self-attention encoder stack (Fig. 2), with the final [CLS] hidden state feeding a linear classification head. Within each transformer block, self-attention is computed over the full sequence of 197 tokens (196 patches + [CLS]) as

$$\text{Attention}(Q, K, V) = \text{softmax}(QK^T / (d_k)^{0.5} V \quad (1)$$

where Q, K, and V are learned linear projections of the token embeddings and dk is the key dimension (64 per head, 12 heads); ViT-Base/16 stacks 12 such blocks, of which the last four are unfrozen in Stage 2 (Section III-C).

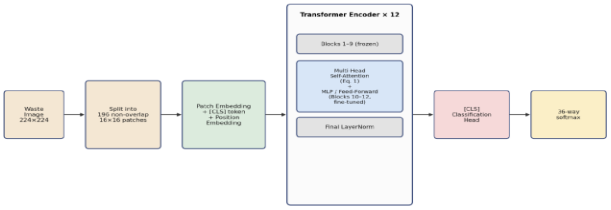

Fig. 2. ViT-Base/16 architecture: patch embedding and [CLS]/position embedding feed a 12-block transformer encoder (blocks 1–9 frozen, 10–12 fine-tuned in Stage 2), followed by the classification head and softmax over 36 classes.

3. Two-Stage Training Procedure

We trained with the Hugging Face Trainer using a custom class-weighted loss (Fig. 3). In Stage 1, the encoder stayed frozen while only the head (27,684 parameters) was optimised for 6 epochs at $lr = 3 \times 10^{-4}$, reaching 90.90% validation accuracy (top-3: 96.91%). In Stage 2, we unfroze the last four encoder blocks, the final layer-norm, and the head (28,380,708 trainable parameters) and fine-tuned for a further 12 epochs at $lr = 2 \times 10^{-5}$, peaking at 95.49% validation accuracy (top-3: 98.07%), which we restored as the best checkpoint. Table I lists the full hyperparameter configuration, and Fig. 4 plots the epoch-wise validation trajectory for both stages.

In both stages, the loss function was class-weighted cross-entropy with label smoothing (ε=0.1); class weights followed the balanced (inverse-frequency) heuristic to offset the dataset's pronounced imbalance — 533 training images for T_Clothes against just 22 for B_Animal Dead Body, for example. Training ran in mixed precision (fp16) on a single NVIDIA Tesla T4 GPU. Formally, for a ground-truth class c over C=36 classes, the label-smoothed target is $y'_i = (1-\varepsilon)y_i + \varepsilon/C$, and the per-example loss is

$$L = -\sum_{i=1}^{C} w_i \, y'_i \log(p_i) \quad (2)$$

where pi is the softmax probability for class i, ε=0.1 is the smoothing coefficient, and $w_i \propto 1/n_i$ is the balanced (inverse-frequency) weight for a class with ni training images, off setting the imbalance between, e.g., T_Clothes (533 images) and B_Animal Dead Body (22 images).

Table 1: Training Hyperparameters

normalisation, upsampling, and blending take over. This mirrors the channel-averaging step of standard Grad-CAM [3], just with the convolutional channel axis swapped for the transformer's hidden-state axis.

4. Deployment as a REST API

The fine-tuned checkpoint and processor are loaded once, at start-up, inside a Flask application with CORS enabled. A POST

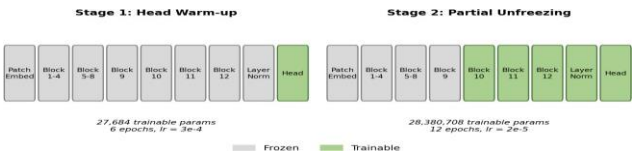

Fig. 3. Two-stage fine-tuning schedule showing frozen (grey) vs. trainable (green) encoder blocks in each stage.

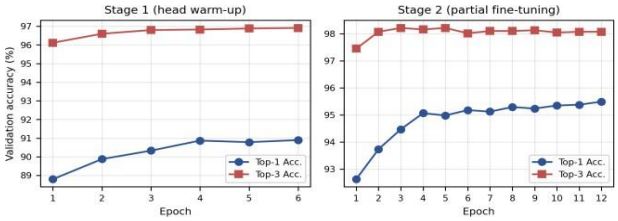

Fig. 4. Validation top-1/top-3 accuracy per epoch for Stage 1 (head warm-up) and Stage 2 (partial fine-tuning).

5. Explainable Grad-CAM for Vision Transformers

Standard Grad-CAM [3] was built for convolutional feature maps, so it doesn't apply directly to a transformer's patch tokens. We adapt it as follows: gradients of the predicted class's logit are back-propagated to the final encoder layer's hidden-state tensor (1×197×768); the [CLS] token is dropped, leaving a 196×768 gradient tensor; per-patch weights are obtained by averaging gradients along the hidden dimension; and a weighted sum over that dimension yields a 196-length relevance vector, which is clipped at zero and min–max normalised. Since ViT-Base/16 divides a 224×224 image into a 14×14 grid of patches, we reshape the vector to 14×14, upsample it bilinearly to 224×224, colour-map it, and alpha-blend it (α=0.4) with the original image (Fig. 1). Formally, for predicted class c, logit yc, and patch-token hidden states A (196×768), the per-patch relevance weight and heatmap are

αp = (1/768) Σd=1768 ∂yc/∂Ap,d,M = ReLU(reshape14×14(α)) (3) for patch index p = 1…196 and hidden dimension d; the ReLU strips out negative, class-suppressing relevance before min–max /predict endpoint accepts a multipart image upload, runs a gradient-enabled forward pass, and computes the softmax confidence of the arg-max class over the 36 logits z:

$$p_i = \exp(z_i) / \sum_{j=136} \exp(z_j) \quad (4)$$

The API returns a JSON payload with the predicted label, $\max_i p_i$ as a confidence percentage, and a base64-encoded JPEG Grad-CAM overlay. Predictions below 50% confidence are relabelled "Unknown Waste" — a deliberate rejection option for open-set deployment. A companion integration-test script exercises the endpoint end to end for continuous verification, and Fig. 6 shows a representative output.

Fig. 6. Original image (left) and Grad-CAM overlay (right) for a cardboard sample.

IV. RESULTS AND DISCUSSION

On the 3,527-image held-out test set, ViT-WasteVision reached 95.66% top-1 accuracy and 98.27% top-3 accuracy, with macro-averaged precision/recall/F1 of 0.93/0.94/0.93 and support-weighted precision/recall/F1 of 0.96/0.96/0.96 (Table II, Fig. 5). It performed best on visually distinctive, well-represented classes such as T_Clothes (F1=0.99, n=533), D_Newspaper (F1=1.00, n=112), and B_Animal Dead Body (F1=1.00, n=22), and worst on classes that were both visually ambiguous and under-represented — notably Z_I_Cigarette Butt (F1=0.80, n=9), Z_F_Smartphones (F1=0.80, n=22), and Z_K_Spray Cans (F1=0.86, n=40). This pattern points to residual class-imbalance effects that even the weighted loss couldn't fully erase.

Table 2: Per-Class Test-Set Performance (36 Classes, N = 3,527)

Table III sets these results against the DP-CNN/En-ELM three-stage benchmark [1], evaluated on the same TriCascade WasteImage collection, along with other recently reported systems. On the full 36-class task, our single-stage ViT pipeline beats DP-CNN/En-ELM's third stage by 10.41 percentage points (95.66% vs. 85.25%) — and does it in one inference pass rather than a three-stage cascade. That gain isn't free: the ViT backbone carries roughly 86M parameters (28.4M updated during Stage 2), against 1.09M for the DP-CNN extractor. The two approaches sit at different points on the accuracy–efficiency frontier — DP-CNN/En-ELM is the better

fit for severely resource-constrained embedded hardware, while the ViT

pipeline makes more sense for server- or edge-GPU-class inference where fine-grained accuracy is the priority.

Table 3: Comparison with Recent Waste-Classification Systems

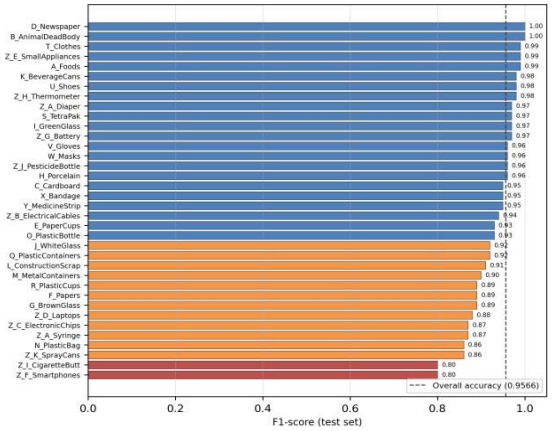

Fig. 5. Per-class F1-score on the held-out test set (36 classes), sorted ascending; dashed line marks overall accuracy.

A qualitative look at the Grad-CAM overlays (Fig. 1 pipeline output) shows the heatmaps consistently tracking the physical extent of the target object — the body of a bottle, the printed surface of cardboard, the cable bundle of an e-waste item — rather than the background, which suggests the fine-tuned attention is genuinely driven by object-relevant features. That lines up with the CNN-based SHAP/Grad-CAM findings reported in [1], even though the two backbones are architecturally very different.

Error Analysis

The confusion patterns in Fig. 5 cluster around materially similar sub-categories: G_Brown Glass (F1=0.89) is most often mistaken for other glass colours, and Z_C_Electronic Chips (F1=0.87) for other small e-waste items — which suggests that separating these classes further would take finer texture cues rather than global shape. Classes with fewer than 25 test images (B_Animal Dead Body, Z_F_Smartphones, Z_I_Cigarette Butt) show the widest gap between precision and recall, a reminder of how much the rarest categories would benefit from more data.

Ablation Study: Effect of Progressive Backbone Unfreezing

The two-stage training schedule from Section III-C doubles as a controlled ablation on how much of the pretrained ViT backbone actually needs fine-tuning to reach strong 36-class accuracy. Table IV isolates this effect, comparing the frozen-backbone configuration (Stage 1: only the 27,684-parameter classification head trained) against the partially unfrozen configuration (Stage 2: the last four encoder blocks, final layer-norm, and head — 28,380,708 parameters) on the same held-out validation split, with the loss function, augmentation, and data held fixed across both rows.

TABLE 4: Ablation on Fine-Tuning Depth (Validation Set, N = 3,526)

Unfreezing the last four encoder blocks adds a further 4.59 percentage points of top-1 accuracy and 1.16 points of top-3 accuracy over the frozen-backbone baseline, at the cost of roughly 28.35M extra trainable parameters and a lower learning rate to avoid catastrophically forgetting the ImageNet-21k features. In other words, classifier-head adaptation on its own isn't enough to specialise general-purpose ViT features to fine-grained waste textures — partial backbone unfreezing turns out to be an effective middle ground between full linear probing and full fine-tuning.

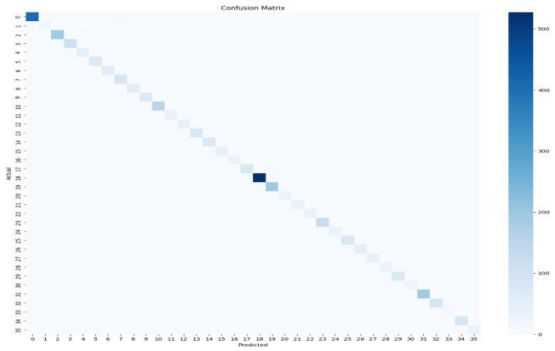

Fig. 7. Test-set confusion matrix (36 classes); diagonal dominance reflects the 95.66% overall accuracy, with residual confusion concentrated among visually similar sub-categories.

The full confusion matrix (Fig. 7) shows a strongly dominant diagonal behind the 95.66% test accuracy, with the handful of off-diagonal entries clustered among the visually similar sub-categories discussed in Section IV-A (glass colours, small e-waste items, and so on). A more complete ablation grid — varying the loss components (class weighting on/off, label-smoothing on/off) and the unfreeze depth (last-2 vs. last-4 vs. last-6 blocks) — would tease these gains apart further; here we report the two-stage result that is actually used in the deployed pipeline and leave the finer-grained sweep for future work (Section V).

Limitations and Future Work

Our evaluation relies on a single stratified split rather than k-fold cross-validation, so the reported metrics carry some unquantified sampling variance. The ViT backbone's parameter count and FLOPs are also substantially higher than the lightweight CNN baselines surveyed in Section II, which could limit deployment on low-power embedded sorting hardware without some form of compression — quantisation, pruning, or distillation. And the current Flask deployment performs synchronous, single-image inference; it hasn't been benchmarked under concurrent load, nor hooked up to a physical sorting actuator, unlike the servo-diverter prototype in [1]. Future work will report cross-validated confidence intervals, explore distilling the fine-tuned ViT into a mobile-friendly backbone, integrate the API with a conveyor/diverter control loop, and collect real-world facility images to cut the reliance on curated web-sourced data.

V. CONCLUSION

This paper has presented WasteVision AI, a Vision-Transformer-based system for fine-grained, 36-class waste image classification, explainability, and deployment. A two-stage fine-tuning schedule — frozen-backbone head training followed by partial unfreezing under a class-weighted, label-smoothed loss — reached 95.66% top-1 and 98.27% top-3 accuracy, beating a recent lightweight CNN-ensemble benchmark on the same source dataset by over ten percentage points on the equivalent 36-class task. A Grad-CAM procedure adapted to the transformer's patch-token hidden states gives pixel-level explanations, and a Flask REST API wraps the trained model, confidence scoring, and explainability output into a deployable inference service that could plug into automated sorting hardware. Where the compute budget allows for it, transformer backbones offer a real accuracy advantage over lightweight CNN-ensemble alternatives for fine-grained waste sorting — with model size remaining the main trade-off for future embedded deployments.

Acknowledgment

The authors thank the maintainers of the TriCascade WasteImage dataset, as well as the open-source Hugging Face Transformers ecosystem that this work builds on.

REFERENCES

1. Md. Nahiduzzaman, Md. F. Ahamed, M. Naznine, Md. J. Karim, H. B. Kibria, M. A. Ayari, A. Khandakar, A. Ashraf, M. Ahsan, and J. Haider, "An automated waste classification system using deep learning techniques: Toward efficient waste recycling and environmental sustainability," Knowledge-Based Systems, vol. 310, art. no. 113028, 2025.
2. A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in Proc. Int. Conf. Learning Representations (ICLR), 2021.
3. R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in Proc. IEEE Int. Conf. Computer Vision (ICCV), 2017, pp. 618–626.
4. T. Wolf et al., "Transformers: State-of-the-art natural language processing," in Proc. Conf. Empirical Methods in Natural Language Processing: System Demonstrations, 2020, pp. 38–45.
5. Q. Zhang, Q. Yang, X. Zhang, Q. Bao, J. Su, and X. Liu, "Waste image classification based on transfer learning and convolutional neural network," Waste Management, vol. 135, pp. 150–157, 2021.

6. A. H. Vo, L. Hoang Son, M. T. Vo, and T. Le, "A novel framework for trash classification using deep transfer learning," IEEE Access, vol. 7, pp. 178631–178639, 2019.
7. S. Neelakandan, M. Prakash, B. T. Geetha, A. K. Nanda, A. M. Metwally, and M. Santhamoorthy, "Metaheuristics with deep transfer learning enabled detection and classification model for industrial waste management," Chemosphere, vol. 308, art. no. 136046, 2022.
8. D. Zheng, R. Wang, Y. Duan, P. C.-I. Pang, and T. Tan, "Focus-RCNet: A lightweight recyclable waste classification algorithm based on focus and knowledge distillation," Vis. Comput. Ind. Biomed. Art, vol. 6, art. no. 19, 2023.
9. S. Jin, Z. Yang, G. Krolczyk, X. Liu, P. Gardoni, and Z. Li, "Garbage detection and classification using a new deep learning-based machine vision system as a tool for sustainable waste recycling," Waste Management, vol. 162, pp. 123–130, 2023.
10. S. M. Cheema, A. Hannan, and I. M. Pires, "Smart waste management and classification systems using cutting edge approach," Sustainability, vol. 14, art. no. 10226, 2022.
11. N. V. Kumsetty and A. Bhat Nekkare, "TrashBox: Trash detection and classification using quantum transfer learning," in Proc. 31st Conf. Open Innovations Association (FRUCT), 2022, pp. 125–130.
12. G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," Neurocomputing, vol. 70, no. 1–3, pp. 489–501, 2006.
13. F. Pedregosa et al., "Scikit-learn: Machine learning in Python," J. Mach. Learn. Res., vol. 12, pp. 2825–2830, 2011.
14. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Commun. ACM, vol. 60, no. 6, pp. 84–90, 2017.
15. S. Kaza, L. C. Yao, P. Bhada-Tata, and F. Van Woerden, What a Waste 2.0: A Global Snapshot of Solid Waste Management to 2050. Washington, DC, USA: World Bank, 2018.