Toward Fully Automated Metal Recycling using Computer Vision and Non-Prehensile Manipulation

Shuai D. Han[1], Baichuan Huang[1], Sijie Ding[2], Changkyu Song[1], Si Wei Feng[1], Ming Xu[3], Hao Lin[4], Qingze Zou[4], Abdeslam Boularias[1], Jingjin Yu[1]

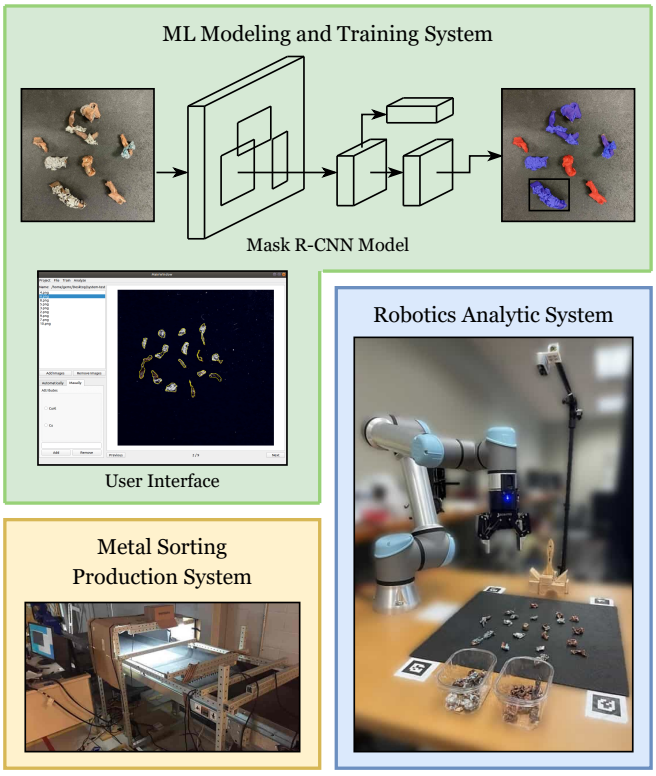
***Abstract*— Due to inherent irregularities in recyclable materials, sorting valuable metals (e.g., aluminum and copper) via mechanical means is a difficult task resisting full automation. A particularly hard challenge in the domain is the separation of scrap metal pieces with physically attached impurities, which is further complicated by variations in different batches of recyclable materials. In this work, leveraging the latest development in machine learning and robot learning, we develop an image-based sorting system for tackling this challenging task. In addition to delivering a highly accurate deep learning model for reliably distinguishing pure scrap pieces from pieces containing impurities with over 95% precision/recall, we further automate the process of sample preparation, data acquisition/labeling/analysis, and machine learning model training.**

I. Introduction

Metal recycling bears with it significant financial and environmental benefits. It allows manufacturers to dramatically reduce the production cost. Using recycled metal, instead of starting from refining raw metal ores, reduces the required energy consumption by 60–95%. At the same time, it leads to a reduction of the associated air pollution by 70–90% and mining waste by 97% [1]. Similar benefits apply to the recycling of many other materials, e.g., plastics.

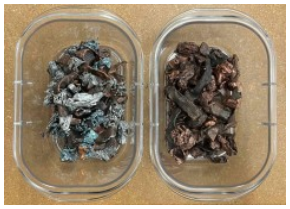
The sorting of scrap metals (specifically, small metal pieces after pre-processing, Fig 1b) is an important step in metal recycling. As recycled materials with metal contents are collected, cleaned and broken into small pieces, the task for a sorting system is to separate different types of metals from the mixed up metal stream. After pre-processed scrap metal pieces are scattered on a moving conveyor, a traditional metal sorting system first uses one or multiple sensors to recognize the pieces and classify their types. The pieces are then localized and moved to different bins (Fig 1c) using some mechanical process, e.g., compressed air jets.

Automatic metal sorting has been a research topic for a few decades [2], [3]. While earlier systems are mostly mechanical [2], [4]–[7], recent works often assume a hardware stack with a conveyor belt that feeds scrap metal pieces, some contact-free sensors to identify and classify the pieces, and a segregation mechanism to eject pieces into different bins. For classification, image-based sensing often directly reasons about the individual RGB channels [8]–[10]. Eddy current based sorting [9], [11]–[14] is effective in separating ferrous metal fractions (e.g., iron, steel) from non-ferrous ones (e.g., aluminum, copper, brass) according to their electrical conductivity and magnetic permeability. Features can also be extracted with Laser Induced Breakdown Spectroscopy (LIBS) [15], [16] and X-ray [17]–[19], and then used in certain classification algorithms [20]–[23]. Some early



(a) System architecture

(b) Mixture of metal pieces



(c) Separated metal pieces

Fig. 1: (a) The three subsystems of the metal sorting system proposed in this article. (b) The scrap metal pieces to be separated, including pure copper pieces and copper pieces with aluminum impurities. (c) Separated pieces.

[1] S. D. Han, B. Huang, C. Song, S. W. Feng, A. Boularias, and J. Yu are with the Department of Computer Science, Rutgers University, Piscataway, NJ, USA. Emails: shuai.han@rutgers.edu, {baichuan.huang, changkyu.song, siwei.feng, abdeslam.boularias, jingjin.yu}@cs.rutgers.edu. [2] S. Ding is with Brown University, Providence, RI, USA. Email: sijie_ding@brown.edu. [3] M. Xu is with GEM Co. Ltd., Tianjin, China, Email: xuming@gem.com.cn. [4] H. Lin and Q. Zou are with the School of Engineering, Rutgers University, Piscataway, NJ, USA. Emails: {hlin, qzzou}@soe.rutgers.edu.
This work is supported in part by a research contract from HSG (Wuhan) Internet Co. and NSF grant IIS-1846043.

works [24], [25] also reason abouts thermal properties. It is fairly common to use a combination of multiple sensing modalities [9], [12], [17], [22], [26], [27].

There are many drawbacks in existing sensor-based sorting systems including mediocre accuracy and poor customizability. For example, current image-based classification methods [8]–[10], [23] are mostly based on RGB pixel-counting which has accuracy at around 80%. They are also proprietary in general and cannot be easily modified. Such drawbacks become quite problematic in metal recycling because purity directly affects the sale value of the recycled metal and different batches of recyclables have different characteristics including colors and shapes. As a prime example, existing methods/systems (image and non-image based) do not function well at all when separating copper pieces from copper pieces with aluminum impurities (Fig. 1b and Fig. 2), a fairly common occurrence in metal recycling. In this work, as a collaborative effort between Rutgers and a large recycling company, we have developed a full hardware-software stack building on the latest machine learning and robot learning research for image-based material sorting. Our method and system can successfully handle the separation of irregular pure copper pieces and copper pieces that are partially wrapped in aluminum.

The system we have developed includes three interacting but independently operating subsystems (Fig. 1a): (1) a machine learning modeling and training subsystem with an intuitive UI for quick data labeling and scheduling training of machine learning models, which can also apply existing models for autonomous pre-labeling, (2) a compact robotic subsystem for analyzing small scrap metal samples, capable of autonomously spreading the samples and sorting the samples using trained classification models, and (3) a metal sorting production system with cameras, conveyors, air jets, and associated control software. Because the main innovations of this work reside with the first two subsystems, this paper and its technical discussion focus on these subsystems.

The main contributions of this work are two. First, as a key part of the overall system, we have developed a high-performance recognition and classification method for image-based metal sorting tasks, building on recent advances in machine learning and deep learning. The method achieves an accuracy (both precision and recall) at above 95%; our method is also highly robust under changing lighting conditions. Second, surrounding the classification method, we have developed a complete and modernized software-hardware stack with a robotic data collection/analytic subsystem, a model training and evaluation subsystem, and a production subsystem. The system allows fast, semi-autonomous analysis and adjustments of the machine learning models to accommodate variations in incoming recyclable materials.

II. Image-Based Sorting of Metal Mixtures

In contrast to making new products from uniform parts, recyclers must deal with incoming recyclables that are highly non-uniform and irregular. Due to the inherent variability in recyclables, the recycling industry faces many challenges in automating their production pipelines for large scale recycling operations. To ground our discussion, in this section, we describe a specific problem we set out to address in this paper: the reliable separation of scrap metals pieces that cannot be sorted using existing technology.

As metal recyclables are processed, large objects are mechanically broken down into smaller pieces that must be sorted based on the type. These metal pieces can be difficult to separate due to two main reasons. First, the mechanical pulverization process can cause different types of metals to get mixed in the same metal pieces. Having impurities in recycled metals results in significantly lowered value. Second, different batches of recyclables can lead to large variations to the color and shape of the resulting metal pieces. For example, just like appliances themselves, the composition of appliance recyclables can change dramatically over time. For notational convenience, we denote the metal sorting problem as Image-based Sorting of Metal Mixtures (`ISMM`).

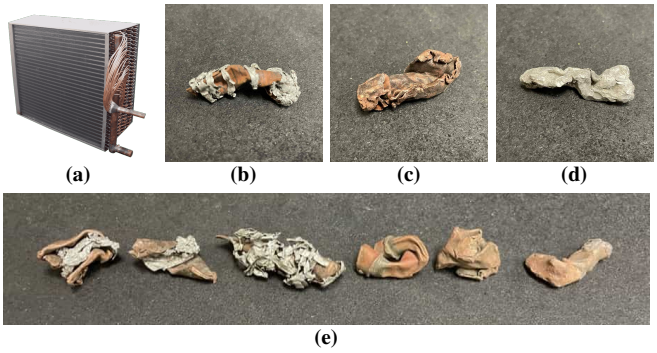


Fig. 2: (a) An evaporator. (b) A copper piece wrapped in aluminum. (c) A pure copper piece. (d) A pure aluminum piece. (e) Three copper pieces with aluminum mixed in and three pure copper pieces. Notice the similarity and difference in their appearances.

As an example, to recycle metals (iron, copper, and aluminum) in evaporators (Fig. 2a) from air conditioners and refrigerators, the evaporators are washed and then pulverized to pieces with the longest dimension around 5 centimeters. Beside copper and aluminum pieces, the mechanical pulverization process creates a significant amount of copper pieces with aluminum impurities wrapped around (Fig. 2b,e). Such pieces are very difficult to separate from the pure copper pieces as they are similar in density and composition in the eyes of existing sorting machinery. Note that, grinding the pieces to be much smaller will separate the aluminum from the copper, but the additional pulverization process also results in significant material loss. Moreover, different batches of evaporators have differences in colors and shapes of the resulting metal pieces. Without an autonomous solution, the problem is currently dealt with manually, which is uneconomical and potentially unsafe for workers, with exposures to sharp metal pieces and loud machinery noises.

For the rest of the paper, we use the copper and aluminum setting as the use case in describing our solution for `ISMM`. We use Cu to refer to pure copper pieces and CuAl to refer to pieces with mixed copper and aluminum content.

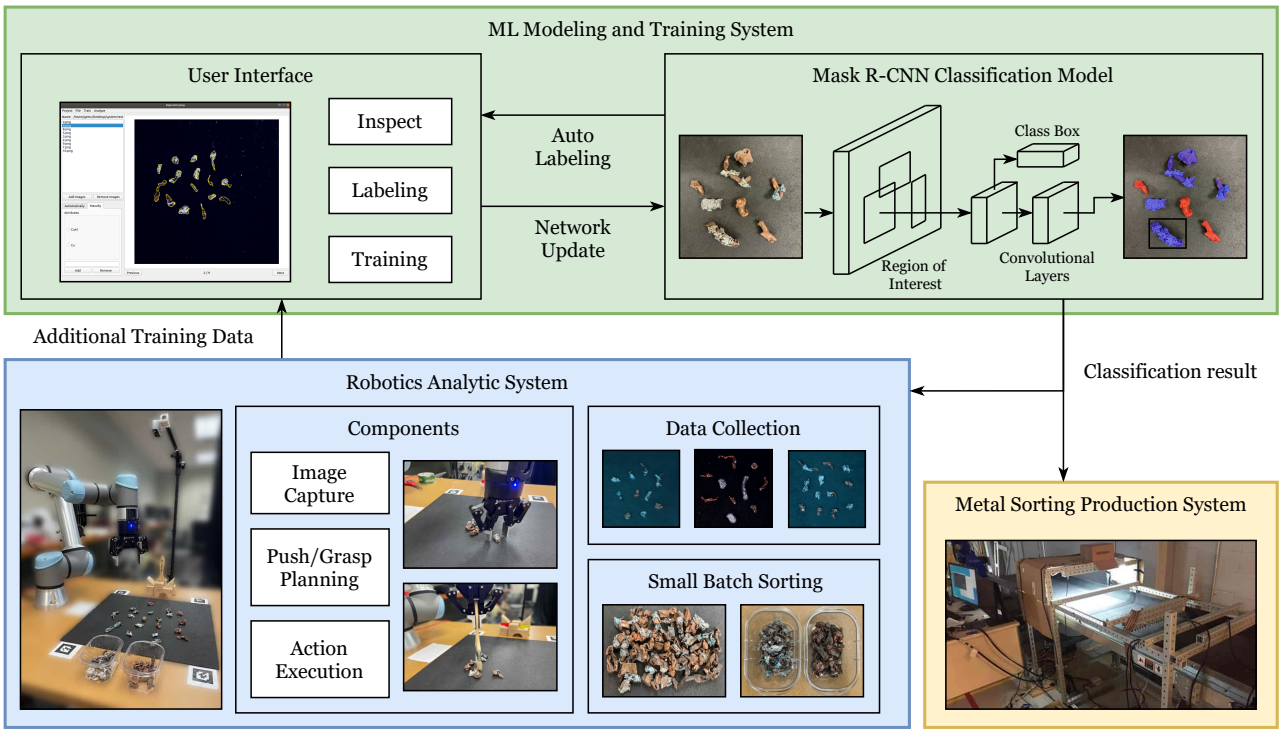


Fig. 3: System architecture. The system has three main components. The ML modeling and training system is composed of a Mask R-CNN [28] classification model, which outputs accurate metal scrap classification result, and a user interface, which is used to inspect and label the training data for Mask R-CNN. To improve the labeling efficiency, the classification model can automatically label the training data for further manual tuning. The second part, named robotics analytic system, employs a camera and a push and grasp policy to handle and analyze small batches of scraps. The robot analytic system can help generate extra training data for the ML model, or directly sort small batch of metal scraps with high accuracy and efficiency. The third part is a conveyor belt production system for large scale metal sorting. Both the robotics and production systems use the Mask R-CNN model to perform object classification.

III. System Architecture

Until recently, no apparent economical autonomous solutions exist for ISMM: image-based and non-image-based solutions cannot reliably tell Cu apart from CuAl due to their similarities. However, rapid advances in machine learning have yielded CNN (convolutional neural network) based Machine Learning (ML) models for reliably discriminating similar objects with levels of accuracy sufficient for solving ISMM. Progress in robot learning further makes it possible to automate a significant part of data collection, analysis, and model training. Based on these observations, we leverage the latest development in machine learning and robot learning to develop sorting methods and systems for ISMM that overcome the shortcomings of existing methods.

As mentioned in Section I, to tackle ISMM, we have developed an effective recognition and classification method. Then, surrounding the flexible and robust method, we have built a complete software-hardware stack comprised of (1) a machine learning modeling and training subsystem, (2) a compact robotic subsystem for automated sample analysis and data collection, and (3) a metal sorting production system. The high-level architecture of the system is given Fig. 3.

In Sections IV, we describe the core recognition and classification methods and the training subsystem. Then, in Section V, we describe the robotic analytics subsystem.

IV. ML Modelings and Training Subsystem

The *machine learning modeling and training subsystem* contains multiple models we have developed for recognition and classification of scrap metal pieces, including one that builds on the the well known Mask R-CNN [28] convolutional neural network model. The subsystem can apply existing trained models for fast classification of training data through its user interface, significantly speeding up the data labeling process. It also has an intuitive user interface for manual data labeling. Here, we first describe the classification models including a color-based model using pixel counting and Support Vector Machine (SVM) [29], and a Mask R-CNN based deep-learning model. Then, we describe the machine learning model training interface. Here, the color-based model, which is a popular choice in industry, is selected as a baseline for comparison purpose.

A. Color-Based Classification

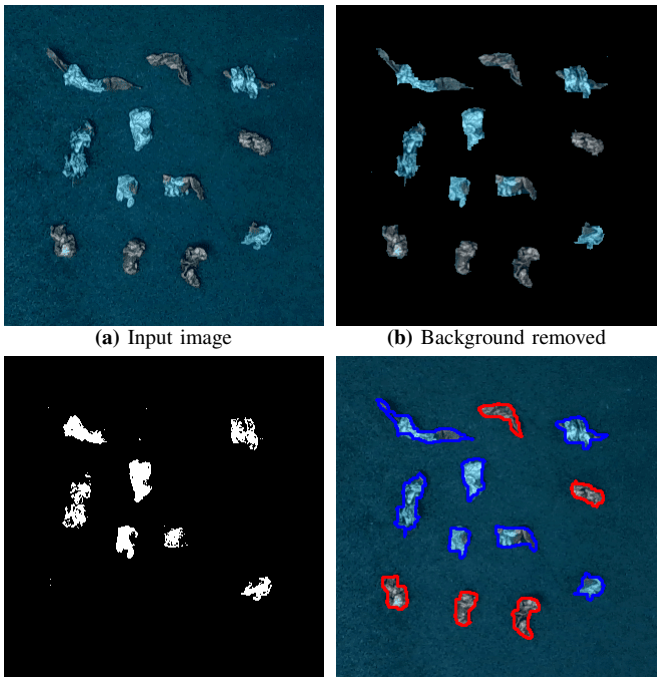
We first develop a baseline color-based classification method that is close to the ones used in the industry [9]. Given an input RGB image (denoted I_{RGB}) captured by a camera above the workspace, the color-based method uses color filters to perform segmentation and classification.

The pseudocode for this method is provided in Alg 1. In line 1, we use dilation and erosion to remove image noise and make the image smoother. Then, in line 2, we apply a filter to remove all background pixels and only retains the pixels that are relevant to the objects. Here, only the pixels with HSV value in some specific range (specified by HSV_{BG}) can pass the filter. An example of the resulting image is provided in Fig. 4b. After another noise removal, in line 4, we retrieve the connected components. If the metal pieces are well-separated

and the background removal is perfect, each $C \in \mathcal{C}$ would correspond to all pixels of one single metal piece. In line 5-8, we check each piece C using a color filter HSV_{Al} (Fig. 4c) to get the percentage p_{Al} of Al in the visible part of the piece, and then use p_{Al} to decide whether the piece is Cu or CuAl. These lines (5-8) can be replaced with other classification methods, e.g., a Support Vector Machine (SVM) model that can be readily trained. An example classification result is provided in Fig. 4d.

Algorithm 1: Baseline Color-Based Classification

1 $I \leftarrow \text{DILATEANDERODE}(I_{\text{RGB}})$
2 $I \leftarrow \text{COLORFILTER}(I, \text{HSV}_{\text{BG}})$
3 $I \leftarrow \text{DILATEANDERODE}(I)$
4 $\mathcal{C} \leftarrow \text{CONNECTEDCOMPONENTS}(I)$
5 **for** $C \in \mathcal{C}$ **do**
6 $\quad p_{\text{Al}} \leftarrow |\text{COLORFILTER}(C, \text{HSV}_{\text{Al}})|/|C|$
7 $\quad$ **if** $p_{Al} > \epsilon$ **then** $t_C \leftarrow$ CuAl
8 $\quad$ **else** $t_C \leftarrow$ Cu
9 **return** $\{(C, t_C)|C \in \mathcal{C}\}$



(a) Input image **(b)** Background removed **(c)** Aluminum pixels **(d)** Result

Fig. 4: (a) An input image of Cu and CuAl scrap pieces. (b) The input image after background extraction. (c) The white pixels corresponds to the aluminum pixels in the original image. (d) The classification result of the color-based method after parameter tuning. The red (resp. blue) lines sketch the contours of Cu (resp. CuAl) pieces. The images are zoomed in for better visualization.

In our implementation, the size for I_{RGB} is 512×512, which reaches a sufficient image clarity. The image smoothing steps (line 1, 3) use 9×9 kernel size. To ensure the color-based system works accurately, the parameters, including the ones for smoothing, the background extraction filter HSV_{BG}, and the material classification filter HSV_{Al}, must be tuned by an expert with computer vision background knowledge, which makes the method difficult to use by non-experts. Apart from manual parameter tuning, another shortcoming for the color-based method is low level of robustness: the accuracy is sensitive to objects' color variation and change of light condition. A slight alternation of such conditions may result in a completely failure of the method and brings the necessity of repeated parameter tuning.

B. Mask R-CNN Based Classification

To overcome the shortcomings of the color-based method, we adapt a deep learning based method that utilizes Mask R-CNN [28]. Mask R-CNN is a flexible and general learning framework for object instance segmentation. It directly works on the raw input image I_{RGB} and is capable of accurately classify all object instances (scrap metal pieces). A high level network structure for Mask R-CNN is provided in Fig. 3: it first finds regions of interest in the image, and then uses a class box for classification and a convolutional network to generate segmentation masks in a pixel-to-pixel manner.

The application of Mask R-CNN has two phases, namely the *training phase* and the *inference phase*. During the training phase, we first take images of metal pieces and then label the pieces inside the images with their respective classes. Such a process is called *labeling*. Note that although it requires manual labor, the amount of images needed is only a few hundreds. Further more, labeling does not require any expertise in computer vision, as opposed to the color-based method where parameter tuning can only be performed by an expert. The labeled images are sent to a training program to automatically generate the Mask R-CNN neural network parameters for inference.

The inference stage is the actual classification process, where the images taken from an running metal sorting system is sent to the trained Mask R-CNN network. The segmentation and classification results are then automatically generated.

C. User Interface for Labeling and Training

To facilitate the deployment and update of the machine learning models, we developed a companion software user interface (see Fig. 5). Its main function is for anyone, including non-expert users, to be able to quickly label sample images for training and updating the Mask R-CNN based ML models. After initialization with pre-trained models, at image loading time, through the UI, an end user can apply existing models (or alternatively, color-based methods) to autonomously label new training images. Then, the end user can manually update any incorrect labels as needed. We have built an easy-to-use tool for creating polygonal object masks for label creation. After a image's labels are verified and saved, the use can schedule background processes to update existing models' parameters to incorporate the newly labeled data. The user interface is straightforward and easy to use. With a basic understanding of the software interface, a human worker without expert computer vision knowledge can readily label at least 30 images in an hour from scratch (i.e., only manual labeling) with around 20 objects per image; the labeling speed can be further increased by several times when automatic labeling is applied.

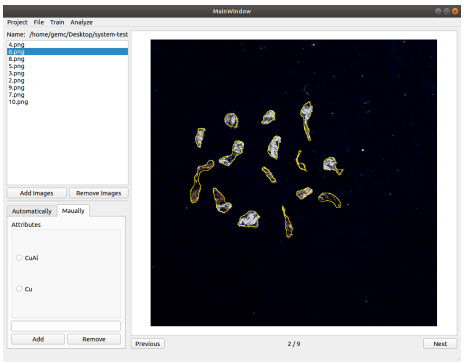


Fig. 5: The user interface for labeling and training Mask R-CNN. The top-left area is used to navigate and manage image data set. The bottom-left area is used for manual and automatic class labeling. User can inspect images and draw labeling polygons with mouse gestures in the right big window.

V. Robotic Analytics Subsystem

Although our classification algorithms have very high precision/recall and are adaptive, as with any deep-learning methods, it needs to be updated as the data distribution changes. In our case, this can happen as a new batch of recyclables is being processed. Therefore, to obtain optimal performance, the models should be updated before each large new batch of recyclables is processed in production.

To provide automated sample processing for facilitating the collection of training data for model tuning, we have designed a compact *robotic analytics subsystem* that can automatically process, analyze, and sort small samples of metal scraps from a batch using the trained classification model. The subsystem is capable of efficiently spreading mixed metal pieces into individual ones. The subsystem can then apply trained recognition/classification models to classify and pick/sort the pieces into separate bins for further analysis. The spread and picking functionality is realized using robot learning (in particular, an adapted Deep-Q Learning Network or DQN [30] is applied). In this section, we present the subsystem from the metal sorting perspective, i.e., the objective is to directly sort different types of metal materials in to different bins. The other functionalities of the subsystem, e.g., object spreading and subsequently data collection, can be directly built on successful metal sorting.

A. Hardware Setup and High Level Workflow

The robotic subsystem (Fig. 6) consists of a manipulator (UR-5 + Robotiq 2F-85), an RGB-D camera (Intel D-435), a square workspace, and control software with an intuitive user interface. With scrap metal pieces arbitrarily placed in the workspace, the robotic system processes the pieces by iteratively performing the following procedures:

1) Sensing: the camera captures an RGB image and a depth image to to get the problem state.
2) Classification: the color image is sent to the machine learning classification subsystem to accurately segment and classify the metal scraps in the scene.
3) Planning: given a set of grasp and push motion primitives and all information from sensing and classification, a DQN is used to select the optimal action.
4) Execution: the robot executes the planned action.

These four steps are repeated until all objects in the workspace are successfully sorted. In the rest of this section, we will describe the planning and execution steps in more detail.

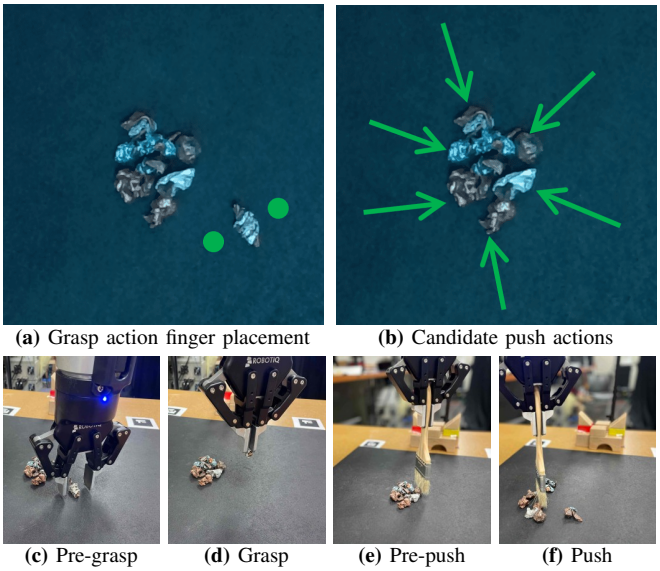
B. Motion Primitives

We use a set of pre-defined motion primitives to avoid on-line motion planning while keeping a high level of action efficacy. Our robotic subsystem has two basic motion primitives: grasp and push, both are executed in an overhead manner, i.e., directly from the top of the workspace.

Grasp actions are designed for the robot to transfer a single object. For a grasp action (x, y, θ, t), the robot hand first moves above the 2D (x, y) location in the workspace with the end-effector pointing straight down, rotated θ degrees along the global z axis (the axis perpendicular to the workspace). Then, the end-effector moves down, closes the fingers, and transports the grasped object to a sorting bin t.

To reach a higher action efficiency and accuracy when processing mixed metal scraps (e.g., see the middle pile in Fig. 6a), we design push actions to help spreading and eventually grasping the objects. For a push action (x, y, θ), we use the end-effector to perform a fixed-distance horizontal sweep motion starting from position (x, y) in direction θ. To increase the contact area and improve the push efficiency, a brush is used. A complete push action includes grasping the brush, moving to an overhead position, performing the sweep action, and then returning the brush to its original position.

The grasp and push actions are visualized in Fig. 6.



(a) Grasp action finger placement **(b)** Candidate push actions

(c) Pre-grasp **(d)** Grasp **(e)** Pre-push **(f)** Push

Fig. 6: Example grasp and push actions for handling metal scraps. (a) The pre-grasp end-effector fingertip positions are shown as green circles. Such a grasp action maximizes the likelihood of picking up the metal piece, while minimizing the chance of fingertip colliding with other objects. (b) candidate push actions that can effectively spread the objects. (c), (d), (e), (f) shows the execution of the grasp and push actions in (a), (b). Notice the single object is successfully retrieved using the grasp action and the scrap pile is well-spread after one push action.

C. Reinforcement Learning of Push and Grasp Strategies

We developed a push and grasp strategy to coordinate and figure out the best grasp and push actions at each iteration.

The strategy is adapted from our related work [30], [31] on object manipulation with *reinforcement learning*, but with a few key modifications on decision making and reward design to customize it to the metal sorting task.

The strategy's outline is provided in Alg. 2. In line 1, we use a Deep Q-Network, named Grasp Network (GN), to predict the optimal grasp action (x, y, θ) and the predicted grasp confidence $\hat{r} \in [0, 1]$ given the current observation, i.e., the RGB image I_{RGB} and the depth image I_{D}. The higher the value of $\hat{r}$ is, the larger the confidence of GN is in the success of the associated grasp action. Then, in line 2, a decision between grasping and pushing is made based on the value of $\hat{r}$: if $\hat{r} < 0.6$, or when the last grasp failed and $\hat{r} < 0.8$, the algorithm chooses to perform a push action. Here, the candidate push actions are generated by uniformly sampling push contact locations on the boundary of objects with push directions pointing to the centers of the objects (see Fig. 6b). Then, a random push action is chosen. When the algorithm chooses to grasp, the target bin t is decided by the class (i.e. Cu or CuAl) of object C at the grasp position (x, y) (line 5). The thresholds used in this algorithm are empirically selected based on the observed performance in the experiments.

Algorithm 2: The Learning-based Manipulation Policy

1 $(x, y, \theta), \hat{r} \leftarrow \text{GN}(I_{\text{RGB}}, I_{\text{D}}, \mathcal{C})$
2 **if** *($\hat{r} < 0.6$) or ($\hat{r} < 0.8$ and last grasp failed)* **then**
3 **return** push action: SAMPLEPUSH($\mathcal{C}$)
4 **else**
5 $t \leftarrow \arg\max_{t_C | C \in \mathcal{C}} |C \cap \{(x, y)\}|$
6 **return** *Grasp,* (x, y, θ, t)

We now provide more background about GN. GN shares a similar structure as the one in [30], which is visualized in Fig. 7. GN focuses its *attention* on local regions that are relevant to each single grasp and finds the optimal policy for maximizing a single-step grasp reward. It takes an input RGB-D image and outputs a score for each candidate grasp centered at each pixel. The input image is rotated to align with the end-effector frame to reason the rotated end-effector (see the left image in Fig. 7). Structural wise, ResNet-50 FPN [32] is used as the backbone; we design the last layer with our own customized head structure shown in Fig. 7. Similar to [30], we also employ image-based self-supervised pre-training to achieve a good initialization of network parameters.

Given the basic structure of GN, we designed the reward function to make the trained policy more suitable to the metal sorting task. For an output grasp action (x, y, θ, t), the possible cases are: *(i)* The grasp is successful and the grasped object is sorted correctly, i.e., the object is put into the correct bin; in this case, the reward is $+1$. *(ii)* The grasp is unsuccessful due to finger collisions and mechanical reasons; in this case, the reward is 0. *(iii)* The grasp is successful but the grasped object is sorted incorrectly; we punish such actions with reward -1. In the experiments, the third case typically happens when the objects are too close to each other, due to sub-optimal pushing actions, which results in a lower classification accuracy and sometimes multiple objects grasped at the same time.

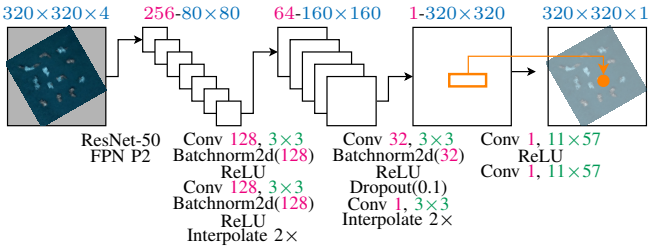


Fig. 7: Architecture of the Grasp Network (GN). Pink, blue, and green text are used for channel count, image size, and kernel size, respectively. The size of the last layer (11×57) is dependent on the end-effector's dimension.

Remark. Prior to the learning-based push-grasping method, we also developed an engineering-based push-grasping method that decides when to grasp and when to push based on a handcrafted heuristic. This engineered method, which does not learn from experience, keeps performing random push actions and only grasps an object when it is classified to be isolated with sufficient clearance. Results of extensive experiments, summarized in the following section, clearly demonstrate the merits of the proposed learning-based approach.

VI. Experimental Results

Extensive experiments were performed to evaluate the classification and robotics subsystems. All systems and methods were trained and evaluated on a machine with an Intel Core™ i7-6800K Processor at 3.40Ghz and an Nvidia GeForce GTX 1070 graphics card.

A. Classification Result on Cu and CuAl

We used the machine learning and modeling subsystem to train the Mask R-CNN and classify Cu and CuAl. We collect data by manually spreading scrap pieces in the workspace and take pictures. A total of 100 images with 168 Cu and 81 CuAl pieces was used for training and parameter tuning, and another 40 images with 61 Cu and 26 CuAl pieces was used for testing and evaluation. Two performance metrics are compared: for a class of objects, *precision* is the number of correctly sorted objects over the total number of objects actually in the class, while *recall* is the number of correctly sorted objects over the total number of objects sorted into in the class. For both metrics, higher is better.

The precision and recall of classifying CuAl is reported in Table I. Rows 2-4 are the baseline color-based methods with classification variations, including pixel counting and support vector machine (SVM) (see Alg. 1). The last row is the result of the proposed subsystem. The results show that Mask R-CNN is much more accurate than the industrial standard color-based methods, with both precision and recall close to 100%; moreover, the gap between the current performance and 100% is caused by human error in labeling and rare occations where objects are not completely scattered.

We report that the Mask R-CNN classification method can process 480p images at around 30 fps or 1080p images

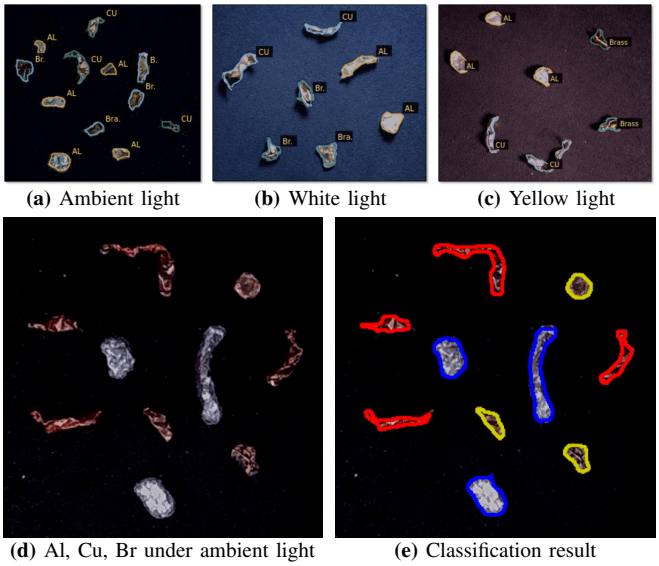
	Precision (%)	Recall (%)
SVM + HSV histogram	95.1	97.9
SVM + RGB histogram	95.0	94.6
Aluminum pixel counting	95.1	96.4
Mask R-CNN	99.3	99.6

TABLE I: Classification Result on CuAl

at around 10fps, a speed that satisfies the requirement for industrial metal sorting tasks.

B. Classification Results under Different Conditions

To show that the proposed system can handle different object shapes, colors, and lighting conditions, we report the classification result with aluminum, copper, and brass metal scraps under various lightning conditions (Fig 8). 300 images were collected to train Mask R-CNN and a total of 96.1% precision and 97.6% recall were achieved for all metal types and light conditions. As a comparison, due to large color variations and similarity between copper and brass, we did not find appropriate parameters for the color-based method to successfully classify this data set.



(a) Ambient light **(b)** White light **(c)** Yellow light

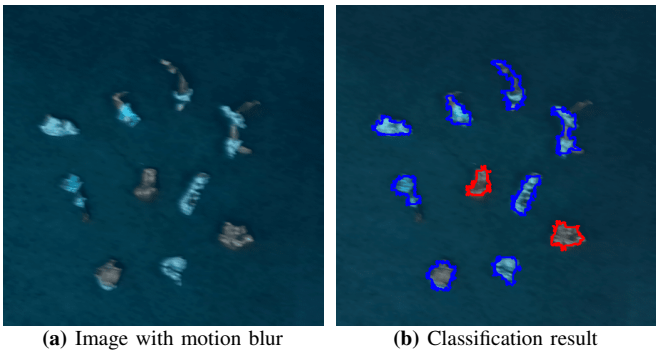
(d) Al, Cu, Br under ambient light **(e)** Classification result

Fig. 8: (a) (b) (c) Aluminum (Al), copper (Cu), and brass (Br) under different light conditions. (d) (e) Example image and classification result.

To test how the method adapts to image artifacts when captured on a moving conveyor belt instead of on a static workspace, we simulate the conveyor belt's movement by mounting the camera on the robotic arm in the robotics subsystem and sweeping the arm above the workspace while taking pictures. The classification result (e.g., Fig. 9) is consistent with what is reported above.

C. Applying the Learning-based Classification Method to a Stator Classification Problem

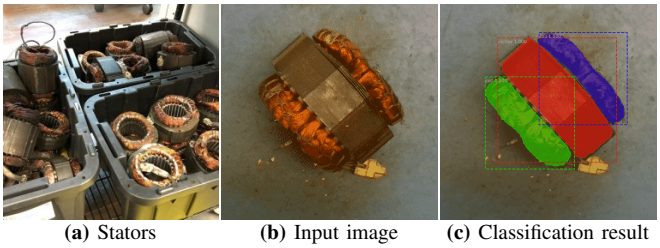
To show our learning-based classification method is easily applicable to new and potentially more complex objects, we apply it to a stator classification problem. In this classification problem, we are given a recycled stator, containing copper coils and silicon steel. The objective is to classify the steel and the coil parts for further processing. An example is shown in Fig 10. The learning-based classification method is trained



(a) Image with motion blur **(b)** Classification result

Fig. 9: The Mask R-CNN model can successfully work with blurry images.

using 100 labeled stator images, and can accurately perform the classification. For this task, the color-based classification method does not work since the object classes are different mainly in terms of texture but could have very similar colors.



(a) Stators **(b)** Input image **(c)** Classification result

Fig. 10: The stator classification problem tasks a classifier to accurately locate the center steel part and the coils of a stator. The stators could vary in terms of size, shape, and color. (a) A collection of stators. (b) A stator is put inside of the workspace. (c) The classification result after training.

D. Experiment on the Robotic Analytics Subsystem

Recall that the main purpose of the robotic analytics subsystem is to automatically process batch samples, spreading them apart, and analyzing the result using existing models. A key measure of its efficiency is the number of actions needed to process a given number of mixed metal pieces (see an example in Fig. 1b). As the grasping portion of the robotic analytics subsystem is learning based, it is also important to examine its data efficiency.

The learning rate of the Grasp Network (GN) is plotted in Fig. 11, which measures the data-efficiency of the learning the grasp policy. We observe that the reward shaping, as detailed in Section V-C, yields fairly good results; the algorithm reaches a fairly high success rate (above 75%) with just around 350 training samples. A 75% success rate means that it takes 1.33 tries to successfully pick up one metal piece from a dense pile of pieces, without any assistance from push.

For the entire push-grasping pipeline, we compare the learning based solution with our initial engineered non-learning-based solution. For the evaluation, we use a total of 100 mixed aluminum and copper pieces (20 pieces per trial) and count the number of push and grasping actions needed for spreading the a pile. Whereas the engineering based method required a total of 239 actions to spread a total of 100 metal pieces, the learning-based method used only 115 actions in total. That is, the robot learning solution only

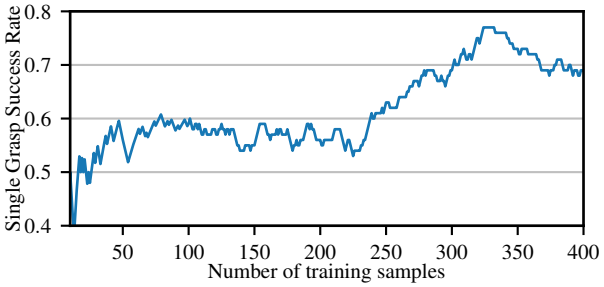


Fig. 11: GN learning curve. The x-axis is the number of grasp actions performed during training. The y-axis is the likelihood to use a single grasp to successfully sort an object.

needs to use 15 actions in addition to the bare minimum of 100 grasping actions to sort and pick the 100 metal pieces. The $100/115 \approx 87\%$ rate is a further improvement over GN, which is an expected outcome.

In the accompanying video of this paper, we demonstrate the functionalities of the user interface (Fig. 5) and a metal spreading clip using the robotics subsystem.

VII. Conclusion

In this paper, we introduce a novel machine learning based metal sorting system comprised of multiple subsystems aimed at high-accuracy separation of scrap metal pieces that are challenging to sort using existing procedures. The system centers around fast and effective classification models that achieve over 95% precision and recall. Surrounding these models, an ML modeling and training subsystem and a robotic analytics subsystem are built to further enhance the robustness and levels of automation of the entire system. Systems like ours bring significant contributions to recycling automation with positive financial and environmental impact, and help free human workers from highly repetitive and labor intensive tasks that also have long-term adverse health risks.

References

[1] U. EPA, "Facts and figures about materials, waste and recycling," *EPA, Environmental Protection Agency*, vol. 12, 2019.

[2] B. E. Anderson and W. C. Larson, "Sorting of scrap metal," June 28 1960, uS Patent 2,942,792.

[3] S. P. Gundupalli, S. Hait, and A. Thakur, "A review on automated sorting of source-separated municipal solid waste for recycling," *Waste management*, vol. 60, pp. 56–74, 2017.

[4] R. R. Osterberg and R. B. Wolanski, "Method and apparatus for sorting non-ferrous metal pieces," May 30 1989, uS Patent 4,834,870.

[5] W. Dalmijn and J. Van Houwelingen, "New developments in the processing of the non ferrous metal fraction of car scrap," Minerals, Metals and Materials Society, Warrendale, PA (United States), Tech. Rep., 1995.

[6] M. Kumagai and Y. Fujita, "Conductive material sorting device," Mar. 7 1995, uS Patent 5,394,991.

[7] K. S. Kimmel, N. A. Hawk, M. A. Keller, and F. Whitmore, "Cullet sorting using density variations," Oct. 15 2002, uS Patent 6,464,082.

[8] P. Kumar, "Metal scrap sorting system," Apr. 8 2003, uS Patent 6,545,240.

[9] M. Kutila, J. Viitanen, and A. Vattulainen, "Scrap metal sorting with colour vision and inductive sensor array," in *International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06)*, vol. 2. IEEE, 2005, pp. 725–729.

[10] T. A. Valerio, "Method and apparatus for sorting metal," Mar. 9 2010, uS Patent 7,674,994.

[11] M. Mesina, T. De Jong, and W. Dalmijn, "Improvements in separation of non-ferrous scrap metals using an electromagnetic sensor," *Physical Separation in Science and Engineering*, vol. 12, 1970.

[12] M. A. Rahman and M. Bakker, "Hybrid sensor for metal grade measurement of a falling stream of solid waste particles," *Waste management*, vol. 32, no. 7, pp. 1316–1323, 2012.

[13] M. Brojboiu, L. Mandache, and V. Ivanov, "Concerning the selectivity of the experimental device based on eddy currents for the metal waste separation," in *2013 4th international symposium on electrical and electronics engineering (ISEEE)*. IEEE, 2013, pp. 1–5.

[14] M. D. O'Toole, N. Karimian, and A. J. Peyton, "Classification of nonferrous metals using magnetic induction spectroscopy," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 8, pp. 3477–3485, 2017.

[15] J. Gurell, A. Bengtson, M. Falkenström, and B. Hansson, "Laser induced breakdown spectroscopy for fast elemental analysis and sorting of metallic scrap pieces using certified reference materials," *Spectrochimica Acta Part B: Atomic Spectroscopy*, vol. 74, pp. 46–50, 2012.

[16] M. Cho, S. Park, E. Kwon, S. Jeong, and K. Park, "A waste metal sorting system using libs classification," in *2019 IEEE 28th International Symposium on Industrial Electronics (ISIE)*. IEEE, 2019, pp. 451–454.

[17] M. Mesina, T. De Jong, and W. Dalmijn, "Automatic sorting of scrap metals with a combined electromagnetic and dual energy x-ray transmission sensor," *International Journal of Mineral Processing*, vol. 82, no. 4, pp. 222–232, 2007.

[18] E. J. Sommer Jr, C. E. Roos, D. B. Spencer, and R. L. Conley, "Method and apparatus for sorting materials according to relative composition," Oct. 14 2014, uS Patent 8,861,675.

[19] D. B. Spencer, J. J. Webster, A. M. Reti, E. J. Sommer Jr, R. E. Hill, and R. L. Conley, "Material sorting technology," Oct. 7 2014, uS Patent 8,855,809.

[20] A. Picón, O. Ghita, P. F. Whelan, and P. M. Iriondo, "Fuzzy spectral and spatial feature integration for classification of nonferrous materials in hyperspectral data," *IEEE Transactions on Industrial Informatics*, vol. 5, no. 4, pp. 483–494, 2009.

[21] S. Koyanaka and K. Kobayashi, "Automatic sorting of lightweight metal scrap by sensing apparent density and three-dimensional shape," *Resources, conservation and recycling*, vol. 54, no. 9, pp. 571–578, 2010.

[22] ——, "Incorporation of neural network analysis into a technique for automatically sorting lightweight metal scrap generated by elv shredder facilities," *Resources, conservation and recycling*, vol. 55, no. 5, pp. 515–523, 2011.

[23] P. Torek, D. F. Gorzen, and K. Chaganti, "Scrap metal sorting system," Nov. 8 2016, uS Patent 9,486,839.

[24] F. Ambrose, R. Brown Jr, D. Montagna, and H. Makar, "Hot-crush technique for separation of cast-and wrought-aluminum alloy scrap," *Conservation & Recycling*, vol. 6, no. 1-2, pp. 63–69, 1983.

[25] R. Brown Jr, W. Riley, and D. Soboroff, "Sorting techniques for mixed metal scrap," *Conservation & Recycling*, vol. 9, no. 1, pp. 73–86, 1986.

[26] H. Kattentidt, T. De Jong, and W. Dalmijn, "Multi-sensor identification and sorting of bulk solids," *Control engineering practice*, vol. 11, no. 1, pp. 41–47, 2003.

[27] T. Takezawa, M. Uemoto, and K. Itoh, "Combination of x-ray transmission and eddy-current testing for the closed-loop recycling of aluminum alloys," *Journal of Material Cycles and Waste Management*, vol. 17, no. 1, pp. 84–90, 2015.

[28] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[29] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

[30] B. Huang, S. D. Han, A. Boularias, and J. Yu, "DIPN: Deep Interaction Prediction Network with Application to Clutter Removal," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2021.

[31] B. Huang, S. D. Han, J. Yu, and A. Boularias, "Visual foresight tree for object retrieval from clutter with nonprehensile rearrangement," *arXiv preprint arXiv:2105.02857*, 2021.

[32] T. Lin, P. Dollár, R. B. Girshick, K. He, B. Hariharan, and S. J. Belongie, "Feature pyramid networks for object detection," *CoRR*, vol. abs/1612.03144, 2016. [Online]. Available: http://arxiv.org/abs/1612.03144