Fault Logging and Assessment for Responsive EV repairs

Ethan Deng ezdeng@ucsd.edu
Jason Gu jig036@ucsd.edu
Daniel Kong dskong@ucsd.edu
Irving Zhao ziz057@ucsd.edu

Mentor: Phi Nguyen Pnguyen@sdge.com
Mentor: Ari Gaffen AGaffen@sdge.com
Mentor: James McCloskey JMcCloskey@sdge.com
Mentor: Anderson Bolles ABolles@sdge.com
Mentor: Kelly Park KPark@sdge.com

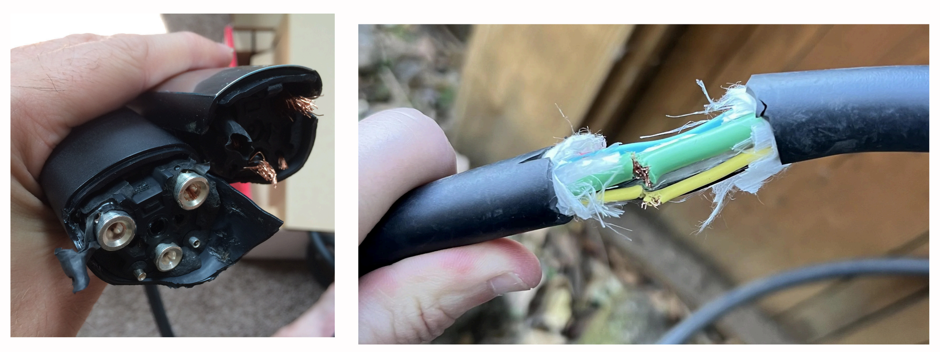
Website:

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE
SDGE

INTRODUCTION

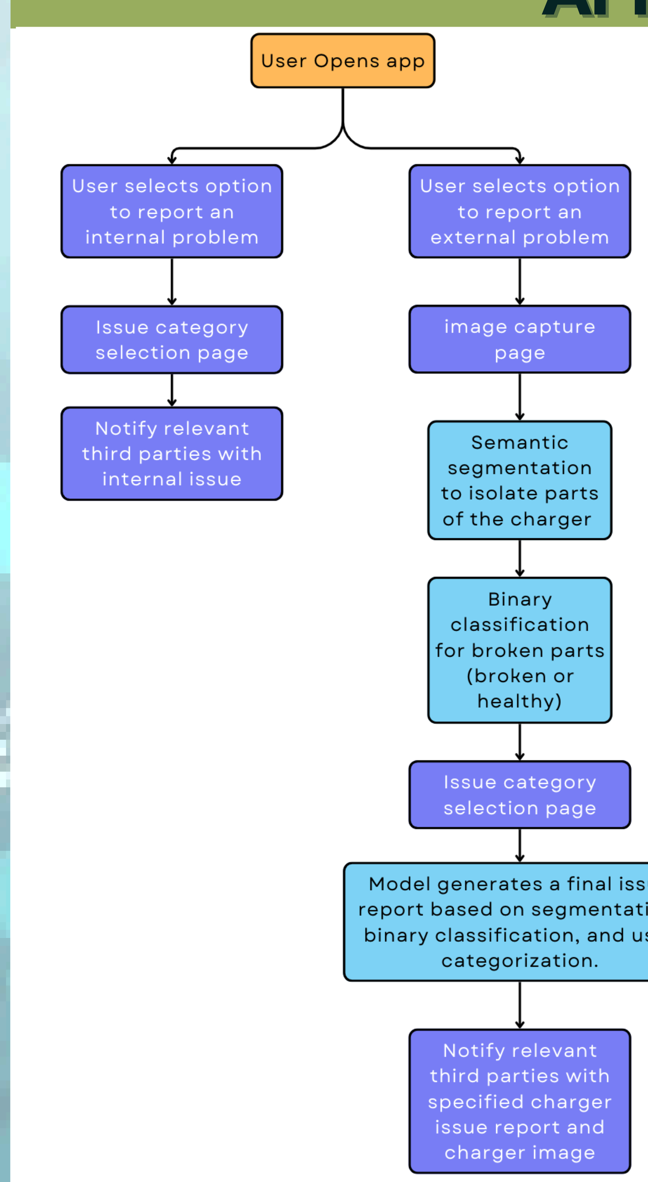
Electric vehicle (EV) adoption is on the rise, increasing the demand for reliable public EV chargers. However, many public chargers are plagued with broken screens, damaged cords, or other charging malfunctions. Users need an efficient reporting system to notify the respective parties of these damages. FLARE aims to address this problem by giving users a platform to submit photos and reports through a mobile app, utilizing our fault detection model to identify specific issues:

- **Damaged cords**
- **Damaged plugs**
- **Housing damages**
- **Broken screens**



FLARE is a proof of concept using semantic segmentation and anomaly detection to detect physical defects in EV chargers.

APP DEMO

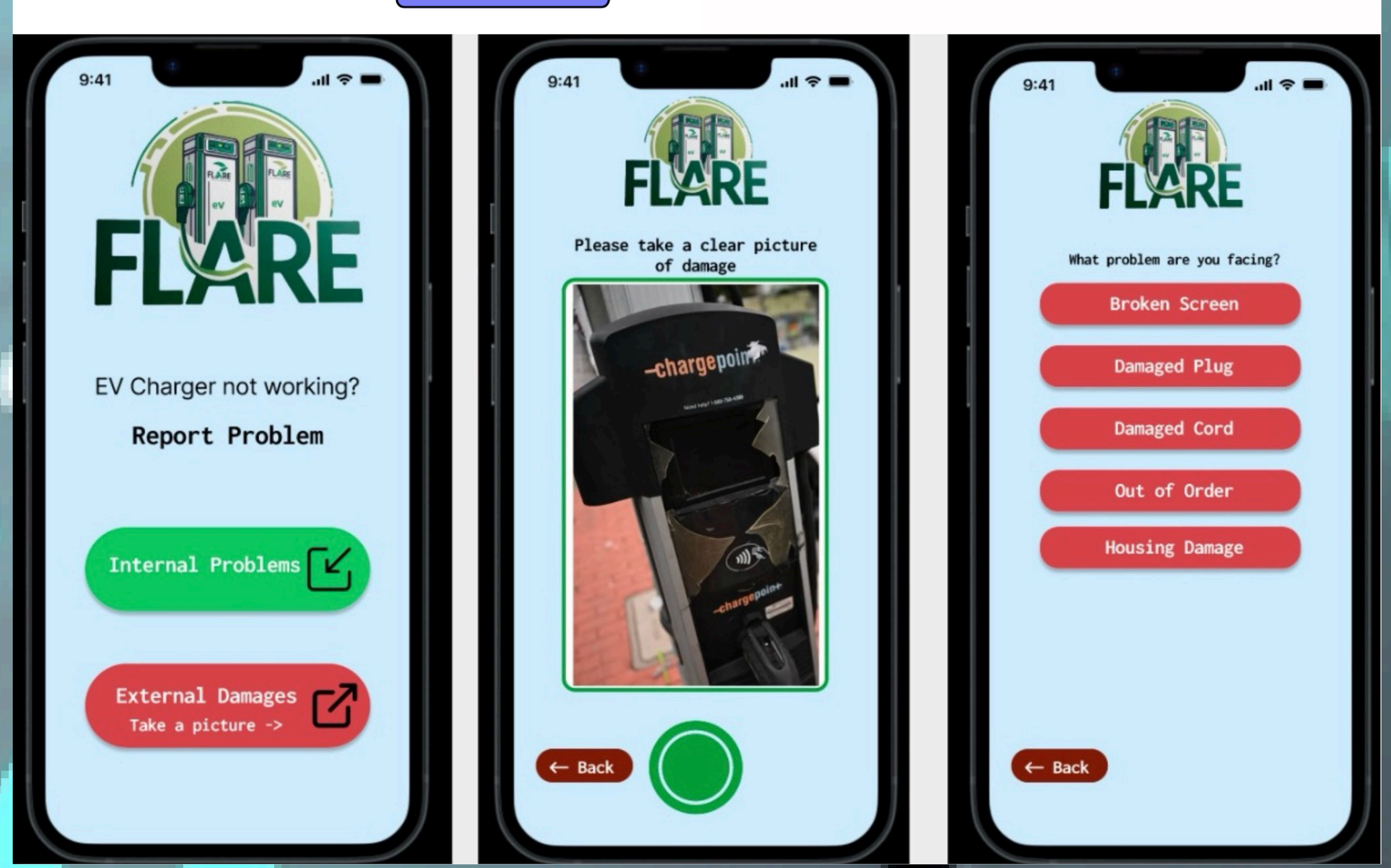


Flow Chart

The operational flow begins with the user selecting an internal or external malfunctioning with an EV charging station. Users that submit a picture of physical damage will have their picture processed by our fault detection model, identifying any issues. Reports are then sent to appropriate maintenance teams for prompt action.

App Interface

Upon launching the app, users are asked to report an internal or external issue. If external, users capture a photo of the charger and select the type of issue from a predefined list.



DATA COLLECTION / PREPARATION

Data Collection

Image data was manually gathered from Reddit, blog posts, news articles, forums, Google Images, and PlugShare. Images consisted of both malfunctioning and healthy charging stations. Additionally, in-person visits to charging locations were conducted to expand the training set, totaling 306 healthy chargers and 101 broken chargers.

Data Annotation and Labeling

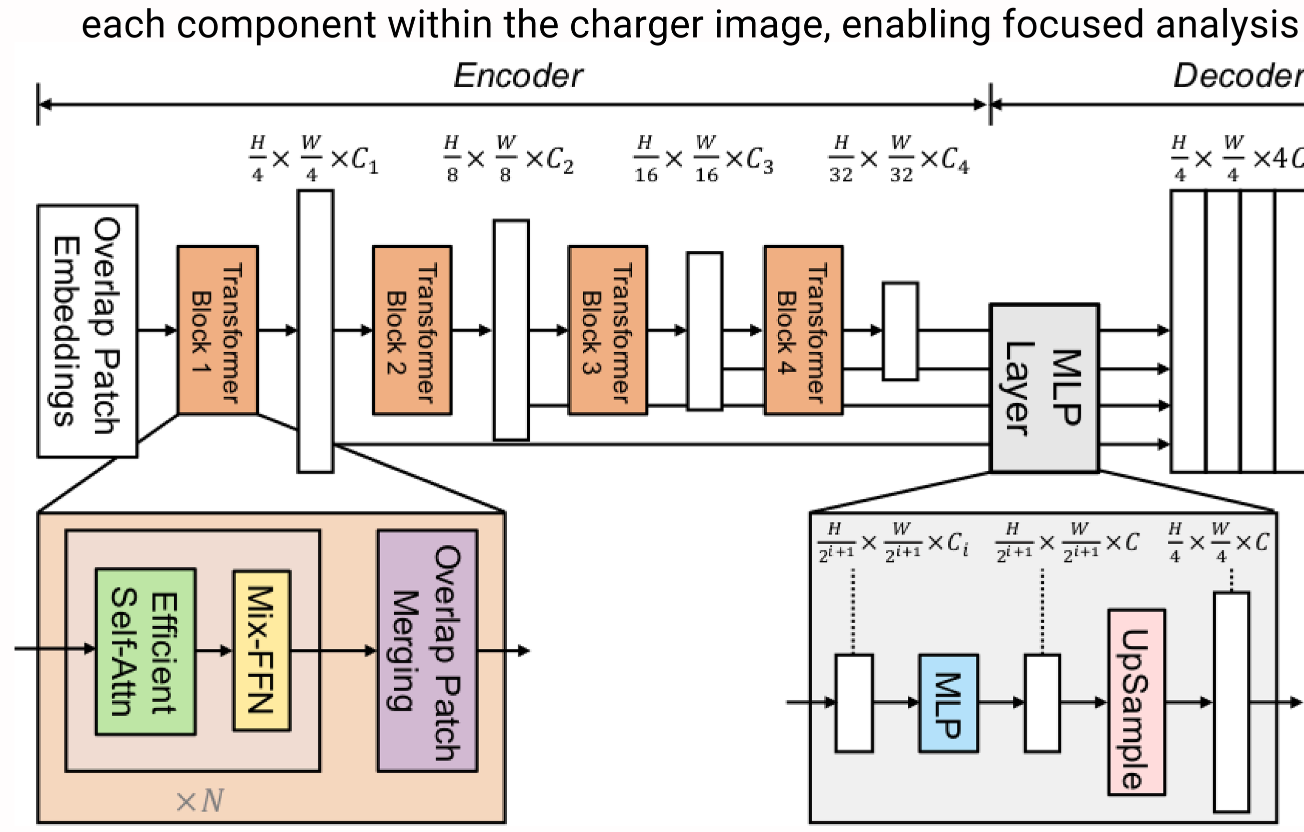
We used Segments.ai, a machine learning-assisted labeling tool, to annotate our dataset. By segmenting the components of EV chargers—screens, cords, plugs, and bodies—we allow our model to recognize parts of an EV charger and determine if each is broken or healthy.

METHODS

To develop a model capable of identifying and classifying EV charger components as either healthy or defective, we executed a two-step process: semantic segmentation followed by binary classification.

Semantic Segmentation - an initial phase segmenting input image into distinct parts.

- Model Selection: We utilized NVIDIA's MIT-B3 segformer architecture, a transformer-based model chosen for its efficacy in semantic segmentation tasks.
- Training Data: Our dataset of healthy and malfunctioning EV charger images is annotated to highlight specific components such as screens, plugs, and cables as described in data collection.
- Training Process: The model was trained and validated to produce segmentation masks that delineate each component within the charger image, enabling focused analysis of individual parts.



The figure to the left illustrates the architecture of SegFormer.

Taken from the "SegFormer: Transformer for Semantic Segmentation" paper.

Binary Classification - components undergo classification to determine condition post-segmenting.

- Feature Extraction: From the segmented regions, features pertinent to the appearance and integrity of each component are extracted.
- Classification Model: We used a vision transformer (ViT) to analyze these features, classifying components as either 'healthy' or 'broken.' This binary classification aids in pinpointing specific issues within the charger infrastructure.

RESULTS

We initially trained a transformer-based model (NVIDIA MIT-B0) on pre-existing datasets (Sidewalk-Semantic) to test feasibility and achieved ~**74.8% accuracy** and a **mean IoU of 0.159** across classes with significantly limited training.

Segmentation

After testing mit-b0 on the sidewalk dataset, we trained an upgraded transformer architecture mit-b3 on our annotated image dataset of chargers through the following:

- Completed 50 epochs of training, 440 steps.
- The model achieved **~87.1% mean accuracy** across classes, **93.5% overall accuracy**, and a **mean IoU of 0.804** across classes after fine-tuning.

Accuracy Screen	Accuracy Body	Accuracy Cable	Accuracy Plug
0.769	0.944	0.771	0.900

Figure 1: Charger Segmentation

Figure 2: Charger Segmentation

Classification: Vision Transformer (google/vit-base-patch16-224-in21k)

FUTURE WORK

Since this project is just the beginning to demonstrate the concept of fault detection for EV chargers, FLARE has many areas for growth to make it truly remarkable.

- **Developing back-end software for the FLARE App:**
  - Integrating our fault detection into an application that users can use to report problems regarding broken chargers.
- **Integrating Plugshare API:**
  - The Plugshare API will be a game-changer for scaling our app to include all the necessary components that make a great EV charger app and attract more users through its greater functionality.
- **Incorporate face and license plate blurring:**
  - To ensure the privacy of private information in user-submitted pictures, model will need to blur faces and license plates before data is collected.
- **Data feedback and continuous training**
  - To provide an increasingly robust model, we plan to integrate user-submitted data into our training pipeline to refine classification accuracy over time - allowing an iterative and inherently scalable approach.

REFERENCES

Strudel, R., Garcia, R., Laptev, I., & Schmid, C. (2021). **Segmenter: Transformer for Semantic Segmentation.** https://arxiv.org/abs/2105.05633

Hugging Face. (2022, March 17). Fine-Tune a Semantic Segmentation Model with a Custom Dataset. https://huggingface.co/blog/fine-tune-segformer

Segments.ai. (n.d.). Documentation. https://docs.segments.ai/

PromptLayer. (n.d.). mit-b0. https://www.promptlayer.com/models/mit-b0-372a