Title:	Self-reconfiguration of a robotic workcell for the recycling of electronic waste
Acronym:	**ReconCycle**
Type of Action:	Research and Innovation Action
Grant Agreement No.:	871352
Starting Date:	01-01-2020
Ending Date:	31-07-2024

Deliverable Number:	D5.6
Deliverable Title:	Use case 3
Type:	Demonstrator
Dissemination Level:	Public
Authors:	Boris Kuster, Matija Mavsar, Gašper Šavle, Martin Bem, Kübra Karacan, Mihael Simonič and Aleš Ude
Contributing Partners:	JSI and TUM

Estimated Date of Delivery to the EC:	31-07-2024
Actual Date of Delivery to the EC:	27-09-2024

Contents

Executive summary

Deliverable **D5.6** is a demonstration deliverable that contains videos exemplifying achievements of the ReconCycle system:

- **Video 1**: Quick layout changes:

  Link: http://reconcycle.eu/videos/d5.6/layout-changes

- **Video 2**: Heat cost allocator disassembly with VLM-based action prediction:

  Link: http://reconcycle.eu/videos/d5.6/hca-vlm

- **Video 3**: Type 1 smoke detector disassembly with VLM-based action prediction:

  Link: http://reconcycle.eu/videos/d5.6/sd-type1-vlm

- **Video 4**: Type 2 smoke detector disassembly with VLM-based action prediction:

  Link: http://reconcycle.eu/videos/d5.6/sd-type2-vlm

- **Video 5**: Opening smoke detector body by unscrewing:

  Link: http://reconcycle.eu/videos/d5.6/unscrewing

The videos demonstrate the system's ability to adapt and reconfigure dynamically based on the specific requirements of each device. The adaptive approach allows the recycling cell to handle variations in device types and physical conditions, ensuring efficient disassembly in diverse scenarios.

1 Introduction

The ReconCycle project aims to develop and implement a flexible and reconfigurable process for recycling electronic waste. We focused on removing batteries from different electronic devices.

In deliverables **D5.1**, **D5.2** and **D5.4** we presented an adaptive disassembly process for different heat cost allocators. By employing the adaptation and self-reconfiguration capabilities of the workcell, we were able to disassemble different devices from the same family. In deliverable **D5.5**, we presented a disassembly workflow for a different family of devices, i.e. smoke detectors, by employing adaptation and self-reconfiguration of the workcell. In this deliverable, we additionally demonstrate:

- a reconfiguration system, allowing for quick workcell layout changes, including automated updating of the digital twin (see Section 2),
- a vision-language model (VLM) based robot action prediction for e-waste disassembly operations (see Sections 3 and 4),
- improved disassembly operations, exploiting tactile information and stiffness adaptation (see Section 5).

Together, the developed system is able to handle both different device families and different device types within the same family, without the need for extensive disassembly line downtimes.

Developing support for quick workcell layout changes required small changes in hardware design, as well as a software package that detects and renders the layout changes in the cell visualization system. Additionally, we developed an approach for adaptive robotic disassembly of electronic waste based on Vision-Language Models (VLMs) and the Retrieval-Augmented Generation (RAG) technique. By combining textual specifications and visual inputs, the system predicts disassembly steps, ensuring logical and context-aware actions through integration with Planning Domain Definition Language (PDDL). This methodology enhances flexibility and accuracy in robotic disassembly operations, addressing the variability of e-waste devices.

2 Quick layout reconfiguration and automatic digital twin updates

In order to support various operations essential for e-waste disassembly, we employ a modular approach where each module is dedicated to performing a specific type of task. Since different device families may require varying sets of operations, it becomes necessary to attach different modules or remove redundant ones.

Video 1 demonstrates the process of removing and attaching a module to the workcell, showcasing the system's modularity.

To enable the use of multiple modules within a reconfigurable cell, it is crucial to integrate the models of all individual modules into the robotic workcell's control system. These models, including geometric data, are utilized for various purposes, including the visualization of the workcell modules in the user interface. This visual representation facilitates operator interaction with the cell, making the system more intuitive and user-friendly. Fig. 1 illustrates the visualization of the workcell both with and without the cutter module.

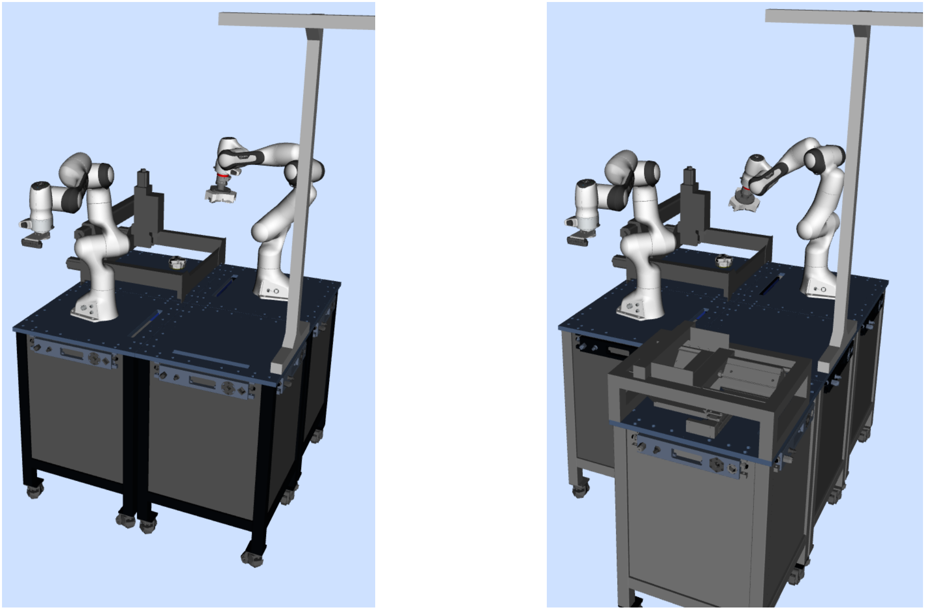

Figure 1: Digital twin of the workcell for two different workcell layouts.

3 Vision-language model based robot action prediction for e-waste disassembly operations

Predicting actions in a robotic workcell is highly context-dependent and requires access to a rich environment representation in addition to sensory information.

To enhance Vision-langauge model (VLM) based prediction accuracy and ensure logical correctness of the predicted steps, we keep track of the environment state in PDDL format. For this purpose, we specify a set of workcell operations, e.g. CNC milling or robotic levering. Our specification includes a textual description that provides specific information about utilizing the desired operation. Using an auxiliary object detection routine based on YOLOv8 [1], we add currently visible e-waste objects and detected relations between them (e.g., "PCB within HCA") to the PDDL environment description. Each adaptive skill is then mapped to a PDDL operator which has preconditions for execution, a set of input arguments/objects, and a set of effects after action execution. This enables the implementation of logical constraints on the permissible actions, such as prohibiting CNC cutting unless an object is clamped within the CNC router. Thus, based on the environment state, a constrained set of valid actions is suggested to the VLM, which ensures that invalid actions can not be predicted and effectively mitigates the hallucination tendencies of VLMs.

After executing a particular action, the action and arguments are appended to a list of executed actions, which is added to the follow-up VLM prompts, enabling the VLM to understand temporal context. If the action has been performed successfully, the effects of each action are applied to the PDDL environment description, which complements the environment description obtained from the vision system. This is beneficial since certain relations can be logically inferred from the executed action outcomes, but may be difficult or error-prone to obtain from vision. For instance, if a cover removal action occurred, the state can be expected to show an open cover.

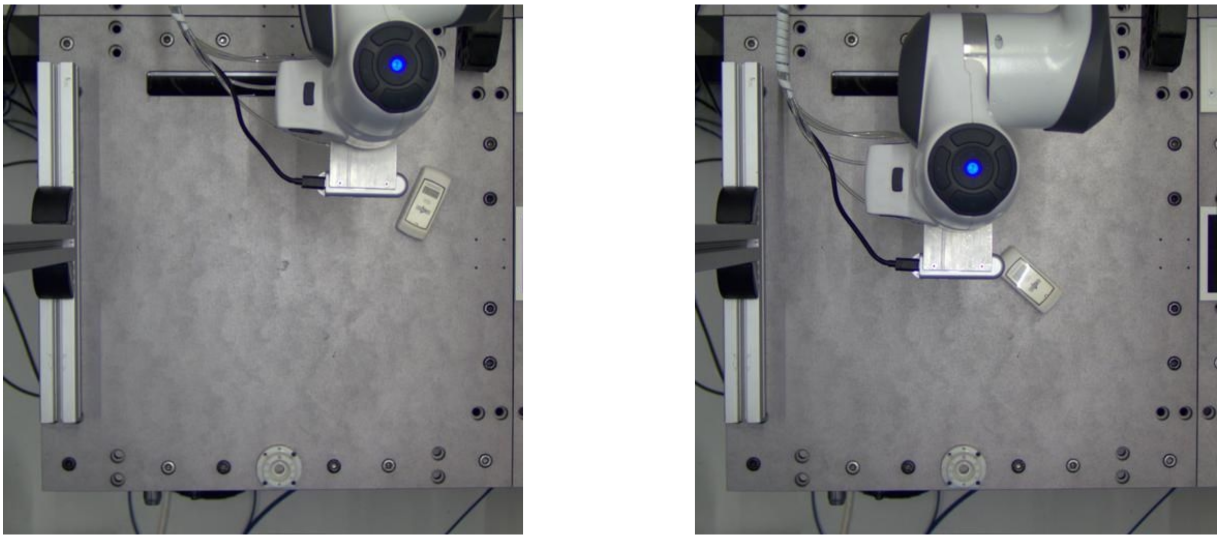

Figure 2: Side-by-side collage: RAG retrieved example on the left, current image on the right.

Improving the accuracy of VLM responses can be further enhanced through Retrieval-Augmented Generation (RAG), a variant of few-shot learning/in-context learning, or by fine-tuning the model. RAG leverages a local database, making it less resource and time-intensive while avoiding the risk of catastrophic forgetting that can occur in fine-tuning. We designed a knowledge database as set of images, where each image has a corresponding textual entry. The textual entry comprises a set of relevant questions and answers, which are both operator-written. The images are embedded into vectors by a visual encoder (we use CLIP [2] in our work) and added to a vector database along with the corresponding texts. The retrieval consists of employing the current camera image (query image) to perform a lookup within the vector database based on cosine similarity between database elements and the query image. Once an image and its corresponding text are retrieved, we combine the retrieved image with the current camera image side by side. The related textual entry is then incorporated into the VLM prompt. In addition to both images, tte final VLM prompt is therefore comprised of:

- A list of (questions, answers) of the retrieved RAG image,
- A list of workcell modules (tables, robots, machines) with short textual descriptions,
- A list of detected e-waste objects,
- A list of detected and known PDDL relations,
- A list of valid actions and their input arguments, as well as brief textual action descriptions.

The process of VLM prompt generation is outlined in Fig. 3 and illustrated with an example in Fig. 4.

The VLM is prompted to respond in JSON format, containing *reasoning*, *action*, and *action_input* fields. The JSON format allows for efficient output parsing. The *reasoning* field, which employs a chain-of-thought prompting style [3], enables the operator to assess the VLM's interpretation of the situation and its proposed course of action. The utilisation of chain-of-thought prompting has been demonstrated to enhance the VLM's performance in scenarios that necessitate intricate problem-solving. The *action* field contains the name of the action to be executed, while the *action_input* field consists of the arguments associated with the aforementioned action. These arguments typically consist of a robot or machine, as well as a candidate object. The proposed prompts are not VLM-specific, but can be used with different off-the-shelf (e.g. gpt-4-turbo via OpenAI API) or fine-tuned open-source models (e.g. LLaVa-1.6-34b, GLM-4V, ...). In our experiments we used the gpt-4-turbo model.

4 Adaptive disassembly workflows for two device families

In deliverables **D5.4** and **D5.5** we presented disassembly workflows for heat-cost allocators and smoke detectors, respectively, where the workcell layout for the disassembly of each device type was different. In this deliverable, we present an implementation that allows disassembly of both device families using the same workcell setup and enables the use of Vision-Language Models (VLMs) to recommend the next disassembly step.

We implement a disassembly workflow, capable of disassembling two different HCA device types and two different smoke detector device types. We additionally utilize a VLM for deciding on the next recommended operation based on the information available at each step.

The experimental setup for the implemented adaptive workflow consists of the following modules:

- *Workpiece Input Module*, where the smoke detectors and heat-cost allocators are supplied.
- *Robot Module* with the Franka Emika Panda robot and a tool changer, enabling attachment of multiple end-effectors (a 3-jaw gripper and a vacuum gripper are utilized in the demonstration).
- *Robot Module* with the Franka Emika Panda robot and the qb Variable Stiffness Gripper (VSG).
- *CNC Module* with a computer-numerical control (CNC) router for the removal of smoke detector housings.
- *Vise Module* with a pneumatic vise for HCA workholding while removing PCBs using levering.
- *Cutter Module* with a pneumatic cutter for the extraction of batteries from HCA devices.

The steps of the general disassembly workflow employing the modules mentioned above are:

- Utilize machine vision to detect locations and types of devices on the input module.
- Transport the object to the corresponding fixture based on the predicted step – a CNC fixture in the case of smoke detectors and a 4-jaw pneumatic vise in the case of HCAs.
- Smoke detectors are cut open using the CNC milling machine, while a levering operation is used for HCAs to extract the PCB along with the battery.

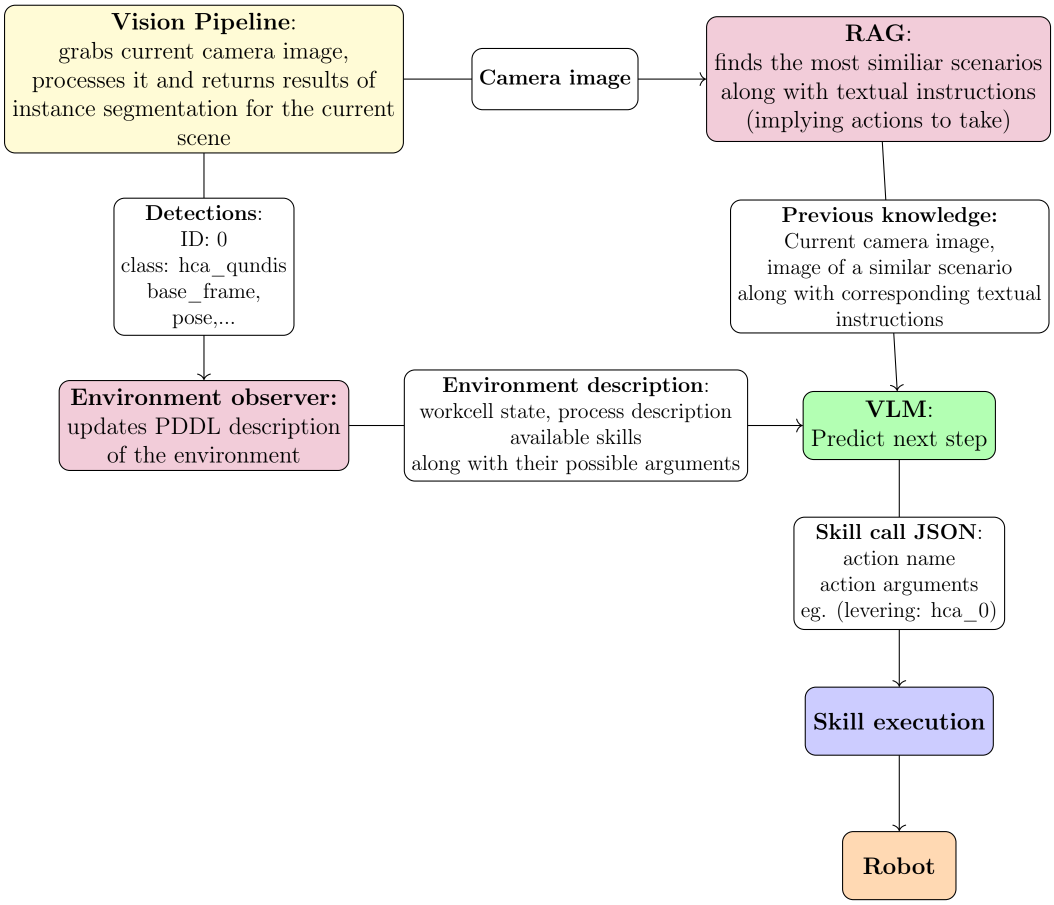

Figure 3: The workflow of a VLM prompt generation. Results from the vision pipeline are passed into the VLM, along with the existing environment description from the PDDL. An appropriate action is performed based on results, obtained from the VLM.

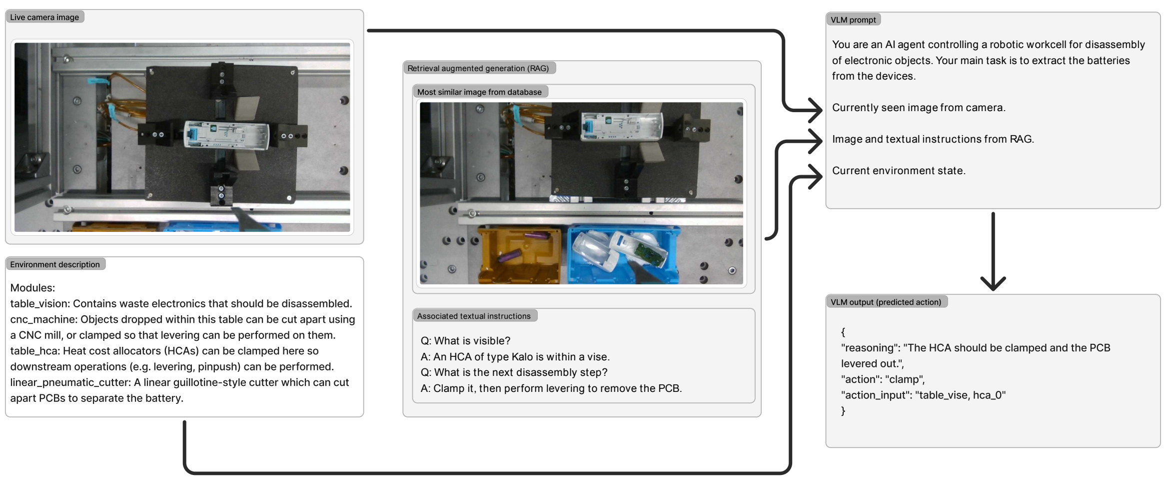

Figure 4: Components of a VLM prompt for action prediction within heat cost allocator disassembly. The predicted action is clamping and subsequent PCB levering.

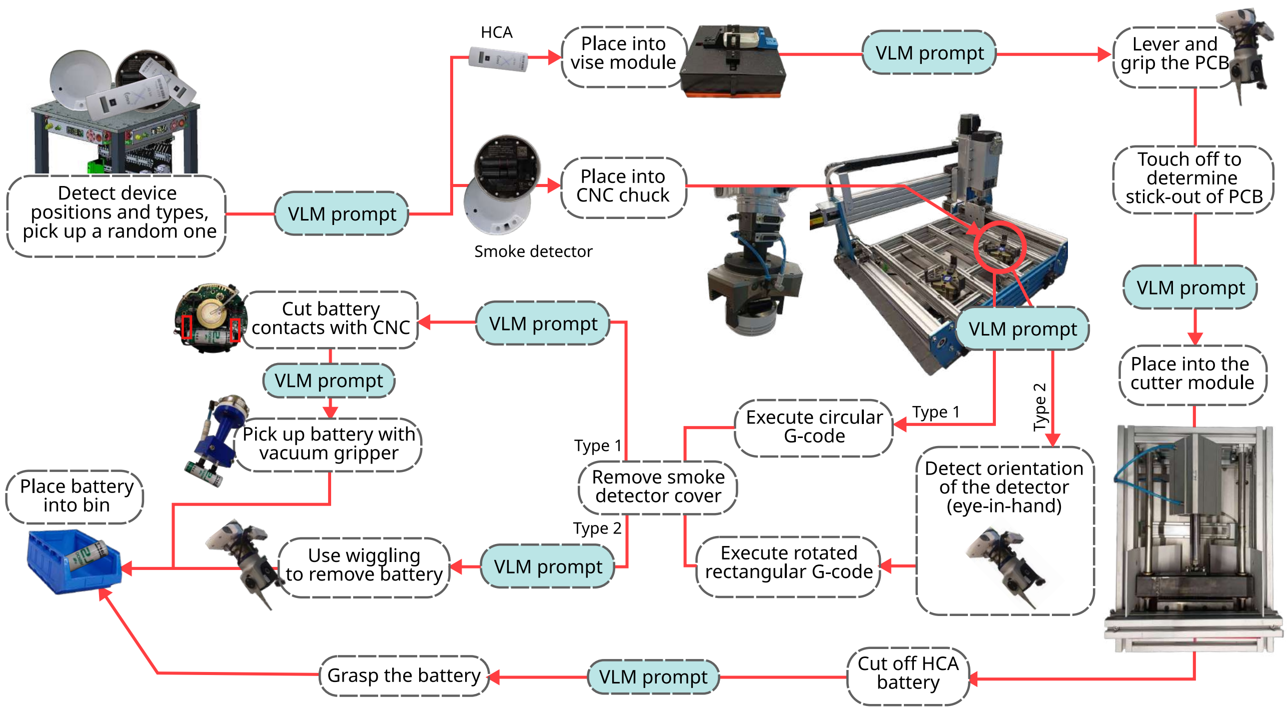

Figure 5: The adaptive disassembly workflow for three different device types (HCA, smoke detectors of Type 1 and Type 2), utilizing VLM predictions for each next step. The flowchart shows the expected disassembly sequences for each device, but other outcomes are possible due to variability in the components and the devices' physical conditions.

- The battery is then extracted – in the case of smoke detectors this includes cutting the battery contacts and lifting the battery, or using a rocking motion to break the contacts, while with HCAs, the battery is cut from the PCB using a pneumatic cutter module.
- The battery is picked up and moved to a storage bin.

A detailed depiction of the expected disassembly scenarios, including VLM prompts, is shown in Fig. 5. The process is demonstrated in **Videos 2-4**.

The VLM is utilized before each disassembly action. The input to the VLM is constructed using the current PDDL environment description, which includes the environment description from the machine vision system, along with additional knowledge obtained from the local RAG database (see Section 3).

5 Opening smoke detector by unscrewing

Dismantling a smoke detector to access the battery involves several coordinated steps, as shown in Fig. 6. First, the system uses a camera to identify the screws and their locations on the detector's cover. The robot then moves to the detected screw location and monitors the force gradient to ensure the precise position of the screw is found. Once the force stabilizes, the screw location is updated, and the robot performs an unscrewing operation for 3 seconds. After unscrewing, the robot moves to an idle position, and a vacuum gripper removes the cover. This process is repeated for all screws. The system utilizes two robots: Robot I, which

Figure 6: The image obtained by the camera including the screws detected and localized and the robot unscrew-driving the detected screw.

is equipped with an RGB-D camera and a vacuum gripper, and Robot II, which is equipped with an unscrewing tool. The procedure is demonstrated in **Video 5**.

6 Conclusion

We presented a hardware and software toolchain for disassembling and removing batteries from heat-cost allocators and smoke detectors, highlighting the system's ability to adapt and reconfigure dynamically based on the specific requirements of each device. This adaptive approach allows the recycling cell to handle variations in device types and physical conditions, ensuring efficient disassembly in diverse scenarios.

Video 1 shows quick layout reconfiguration using plug-and-produce connectors, which enable to quickly attach modules, that are necessary for a different process, or remove unnecessary modules. The layout changes are reflected in the digital twin. **Videos 2-4** show disassembly cycles for three different devices, where a VLM selects the appropriate disassembly operations, which are adapted based on vision results. Several disassembly actions are employed to expose the device's battery, remove it from the device and move it to a container. **Video 5** shows an alternative solution of opening a smoke detector by unscrewing instead of using the CNC milling operation to cut the device open. While not always feasible, unscrewing is a less dangerous opreation than CNC cutting, which can harm batteries contained in electronic waste if applied incorrectly.

Our analysis has shown that it is quite difficult to open smoke detectors, as they are generally designed not to be user-serviceable. Thus, to facilitate the recycling of smoke detectors, the manufacturers of smoke detectors should redesign their products. More specifically, it would be beneficial if the batteries were designed to be removable, preferably with a spring-type base, instead of being soldered in place. This would enable an efficient separation of smoke detector components and streamline the recycling process. Another possibility for improvement is the design of smoke detector covers, which are currently very difficult to remove with robots.

References

[1] G. Jocher, A. Chaurasia, and J. Qiu. *Ultralytics YOLO*. Version 8.0.0. Jan. 2023. URL: https://github.com/ultralytics/ultralytics.

[2] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. *Learning Transferable Visual Models From Natural Language Supervision*. 2021. arXiv: 2103.00020 [cs.CV].

[3] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou. *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models*. 2023. arXiv: 2201.11903 [cs.CL].