Semi-Supervised Approaches to the Video Mirror Detection Problem

Iain High

4th Year Project Report
Artificial Intelligence
School of Informatics
University of Edinburgh

2024

Abstract

Existing mirror detection datasets have high similarity between videos in the training and test set. This leads to existing video mirror detection models performing reasonably well on the test set. However, this performance isn't observed on other datasets that consist of dissimilar data. In other words, the model doesn't adapt well because of the dataset similarity.

To address this issue, we introduced a large unannotated dataset consisting of 219,053 video frames and a smaller labeled dataset to measure the model's adaptability. We measured the cosine similarity between the existing dataset, and the newly created datasets to quantify the issues of data similarity. We also extend the existing SOTA model to introduce three different semi-supervised techniques - namely self-training, expectation maximisation, and co-training - during model training to enable the model to learn from the more diverse data.

Using these semi-supervised techniques on the more diverse dataset, we find that our new approaches significantly outperform the existing SOTA model on the newly created annotated dataset. This signifies that the model has better adaptability. On top of this, we also find that our semi-supervised training approach slightly outperforms the existing SOTA on the existing test set. Once again this signifies that not only do our approaches adapt better to unseen data - they also get a better understanding of video mirror detection as a whole.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Iain High*)

Acknowledgements

This work is dedicated to my father, George High. Who sadly passed away on the 7^{th} of April 2024. His unconditional love and support was a great source of strength during my time at university. My dad was the greatest man I've ever known.

Thank you for everything you've done for me. I'm eternally grateful to call you my dad.

Table of Contents

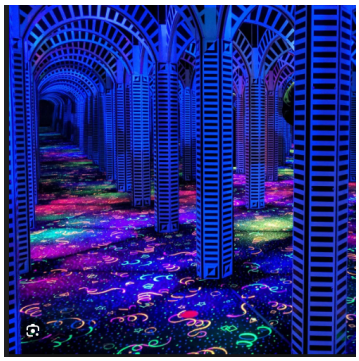
Chapter 1

Introduction

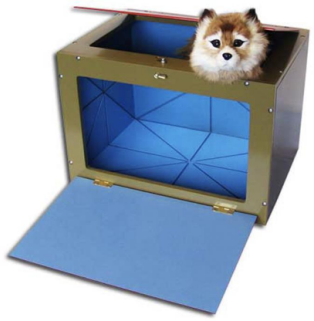
1.1 Motivation

Mirrors and reflective surfaces pervade our modern environment, posing a unique and often overlooked challenge within the field of computer vision. Their variable appearances, heavily influenced by the surrounding environment, make them particularly elusive both to state-of-the-art computer vision models and even to the human eye. Notoriously, they can frequently deceive humans such as in cinematography, magic, and mirror mazes as can be seen from figure 1.1.

Sucker Punch Mirror Scene



Maze of Mirrors



Magician Box

Figure 1.1: Challenging scenarios where humans can't easily identify mirrors: a magician box (right), a mirror maze (middle), and a scene from Zack Snyder's 2011 movie "Sucker Punch" (left) where two actresses are often mistaken for a mirror reflection.

Despite their significance, mirrors remain absent from major object detection and semantic segmentation datasets such as ImageNet Deng et al. (2009), CIFAR Cif, and COCO Coc. This gap underscores the necessity for enhanced models that can reliably navigate the challenges posed by mirrors and reflective surfaces in everyday scenarios.

The nuanced complexity of mirrors significantly impacts various computer vision tasks. Zendel et al. (2017) highlighted the presence of mirrors in datasets as a notable hazard. Corroborating on this are the findings of Braun et al. (2019), who identified reflections

as one of the six primary error sources in person detection. Further emphasising their disruptive presence, Anderson et al. (2018) identified mirrors as potential impediments to vision-and-language navigation (VLN) tasks. General detection methodologies, including depth estimation Tan et al. (2021), vision-and-language navigation Anderson et al. (2018), and semantic segmentation Zhou et al. (2017), often falter in accurately identifying mirrors. Ignoring these mistakes in computer vision tasks may cause severe safety issues in situations such as drone and robotic navigation. Thus, it is necessary to build a robust computer vision model that can distinguish between mirrors and their surroundings correctly.

The majority of mirror detection focuses on identifying mirrors in single images. These models leverage cues, such as contextual contrast Yang et al. (2019), explicit correspondences Lin et al. (2020), semantic association Guan et al. (2022), and chirality Tan et al. (2023). Despite these recent efforts into the mirror detection problem, there's only one model that focuses on mirror detection in videos - VMD-Net Lin et al. (2023).

The video mirror detection problem is important because many real-world computer vision applications are video-based, including robotic navigation, autonomous driving, and surveillance. Addressing the video mirror detection problem can significantly benefit these applications. The dynamic nature of videos presents unique challenges and additional information that are not available in static images. For example, optical flow information and long-term dependencies. In summary, the video mirror detection problem is of crucial importance and is relatively unresearched. This leads to opportunities to drastically increase the performance over existing SOTA models.

1.2 Contributions

The current SOTA mirror detection model VMD-Net Lin et al. (2023) faces many issues due to the Video Mirror Detection Dataset (VMDD) used to train the model. Specifically, this dataset has a high degree of similarity between videos in the training and test set. This lead the model to high performance on the test set, however, when tested on another annotated dataset, the performance is considerably worse - the model doesn't generalise well to videos that are different from that found in training.
To resolve this issue and improve the model, we aim to utilise various semi-supervised approaches to improve the generalisation accuracy of the model. Specifically, our project is composed of three main steps:

1. Firstly, Collecting two new video mirror datasets - a large unannotated dataset used for semi-supervised training, and a small annotated dataset used to measure the adaptability of the model.
2. Secondly, extending the existing SOTA video mirror detection model VMD-Net to use semi-supervised learning by incorporating self-training, expectation-maximisation (EM), and co-training.
3. Finally, testing each of these semi-supervised approaches on both the VMDD-train and newly created annotated dataset to measure performance and adaptability.

4. All code used in this project - including code to generate the new unlabeled dataset - can be found on my GitHub: https://github.com/IainHigh/HonoursProject/tree/main
5. With minimal annotation effort we were able to significantly improve the SOTA model by using semi-supervised learning to address the data similarity issues.

1.3 Dissertation Structure

This dissertation comprises of five main chapters presenting important information about the implementation of this project.

Chapter 2: This chapter explores the relevant literature that our project builds off of. It explores both the history of mirror detection techniques as well as the current SOTA methodology.

Chapter 3: This chapter discusses the dataset in more detail. Exploring how we quantised the dataset similarity, and explaining how we created the two new datasets used in this project.

Chapter 4: This chapter discusses the implementation of each of the three semi-supervised approaches. It also covers the evaluation metrics used to analyse the performance of the models on the two test sets as well as the hyperparamers used in model training.

Chapter 5: This chapter discusses the results obtained through our semi-supervised approaches and compares them to the baseline of the SOTA model.

Chapter 6 This chapter concludes and evaluates our work. It identifies the limitations of our approach and presents future projects that would expand this work. There's also a brief mention of the live demo used at the University of Edinburgh School of Informatic's project day.

Chapter 2

Background

2.1 Difficulties in Mirror Detection

Mirror detection presents unique challenges for computer vision systems, including a variety of widely used models such as R-CNN Girshick et al. (2014), YOLO Bochkovskiy et al. (2020), and Mask R-CNN He et al. (2017). The intrinsic properties of mirrors, coupled with environmental variability, make them difficult to detect for standard detection algorithms. This section explores the multifaceted difficulties encountered in mirror detection.

1. Mirrors, by nature, do not possess a fixed appearance but instead reflect their surroundings, making their visual content highly variable and dependent on the environment. This characteristic alone significantly complicates the detection process, as object detection models will tend to detect the objects in the mirror instead of the mirror itself.

2. The lack of inherent, distinctive features in mirrors further exacerbates the problem. Unlike objects with specific textures, patterns, colours, or shapes (e.g. vehicles, animals, furniture), mirrors exhibit the properties of the objects and scenes they reflect. This absence of unique features makes it challenging for computer vision models to learn and identify mirrors based solely on appearance.

3. Boundary ambiguity is another critical issue. In cases where mirrors have minimal or no frames, distinguishing the mirror's edge from its surroundings becomes difficult. This ambiguity complicates the task of defining the exact limits of the mirrors, essential for pixel-level segmentation. For example, the middle image in figure 2.4 shows a borderless mirror on a white wall reflecting another white wall.

4. Mirrors also introduce complexities related to perspective and depth. The reflection in a mirror might portray objects as being located behind the plane of the mirror, reversing their actual depth. This reversal can lead to errors in algorithms that rely on depth information for object detection and scene understanding.

5. The varying illumination conditions reflected in mirrors pose additional challenges for detection algorithms, which must be capable of generalising across a wide

range of lighting scenarios. This requirement for versatility in handling different illumination conditions adds another layer of complexity to mirror detection.

6. A significant hurdle in developing robust mirror detection models is the scarcity of specialised training data. Capturing and annotating a dataset that adequately represents the diversity of mirrors and their reflective behaviours in numerous contexts is a time-consuming task. Some datasets have been collected such as the Mirror Detection Dataset (MSD) Yang et al. (2019), Progressive Mirror Dataset (PMD) Lin et al. (2020) and Video Mirror Detection Dataset (VMDD) Lin et al. (2023). However, compared to modern deep-learning datasets, these are all relatively small.
7. Occlusions and partial views further complicate mirror detection. Mirrors might be obscured by other objects or only partially visible from the observer's viewpoint.
8. The diversity in the optics of the mirrors can present additional challenges. Mirrors can be flat, concave, or convex. Each introduces unique reflections and distortions that must be accounted for by detection algorithms. Our project specifically focuses on flat mirrors.
9. Surfaces that mimic the reflective properties of mirrors, such as polished metals or water bodies, can lead to false detection. Distinguishing genuine mirrors from other reflective surfaces requires careful analysis and often additional contextual information. A general reflection detection project has been proposed in the future work section in chapter 6, but our project is only interested in detecting mirrors.
10. Finally, Distinguishing between mirrors and windows is also notably challenging due to their optical similarities. Without in-depth scene understanding, identifying one from the other is difficult. Mirrors reflect the environment in front of them, while windows offer transparency behind them. Because of this, the reliance on apparent depth information leads to mirror detection usually detecting windows as well. Advanced algorithms that analyse contextual cues, spatial relationships, and reflection characteristics are essential for accurately distinguishing between these two types of surfaces.

Given these complexities, it is clear why mirror detection has evolved as a specialised domain within computer vision. The challenges outlined above necessitate innovative approaches, often requiring the development or significant adaptation of existing models to effectively address the unique characteristics of mirrors.

2.2 Semantic Segmentation

Object detection serves as a foundational task in computer vision, aiming to identify and locate objects within images or videos. This process involves two primary steps: first recognising the presence of objects across various classes, and second, pinpointing their spatial locations typically via bounding boxes. Each detected object is assigned a class

label along with a bounding box that outlines its position within the image, facilitating the distinction between different objects and their precise locations.

Semantic segmentation advances beyond object detection by assigning a class label to each pixel in the image, thereby providing a more granular understanding of the scene. Unlike object detection, semantic segmentation does not differentiate between individual instances of the same class. This means that if an image contains multiple objects of the same type, such as two dogs, semantic segmentation treats them uniformly, assigning the same class label to every pixel belonging to any dog. This approach is pivotal for applications requiring detailed analysis of the image composition, including the identification of object boundaries.

Instance segmentation represents a more sophisticated approach that combines the principles of object detection and semantic segmentation. This method not only labels each pixel according to its class but also distinguishes between different instances of the same class. Consequently, in an image featuring two dogs, instance segmentation identifies and labels each dog separately, enabling the recognition of individual objects and their specific contours. This hybrid approach offers a comprehensive analysis allowing the detailed segmentation of objects while preserving their distinct identities.

In the context of this research, the focus is on semantic segmentation - identifying mirror surfaces and assigning each pixel one of two classes (mirror and non-mirror). Given the singular class interest, the task does not necessitate distinguishing between multiple mirrors. Therefore regardless of the number of mirrors present in the scene, all will be uniformly labeled under the same class. This approach aligns with the semantic segmentation methodology, where the primary objective is to accurately classify each pixel as either part of a mirror or not, without distinguishing between individual mirror instances. This simplification is crucial for the targeted analysis of mirror surfaces, facilitating the development of models capable of recognising and delineating mirror areas effectively within diverse visual contexts.

2.3 Previous Work in Mirror Detection

2.3.1 General Segmentation Models

One of the biggest changes in the field of computer vision is access to extremely large datasets that allow the training of generalised object detection models. These models have been found to perform very well over a wide range of classes. Examples of these are OpenAI's CLIP Radford et al. (2021); Amazon AWS Rekognition AWS; and Meta AI's Segment Anything Model (SAM) Kirillov et al. (2023). We will be discussing Meta AI's SAM as it focuses on segmentation as opposed to object detection using bounding boxes so we can make comparisons to the specialised image-based models for detecting mirrors.

Segment Anything Model Kirillov et al. (2023) is a segmentation system with zero-shot generalisation to unfamiliar objects and images, without the need for additional training. It was trained with more than one billion masks collected on eleven million licensed

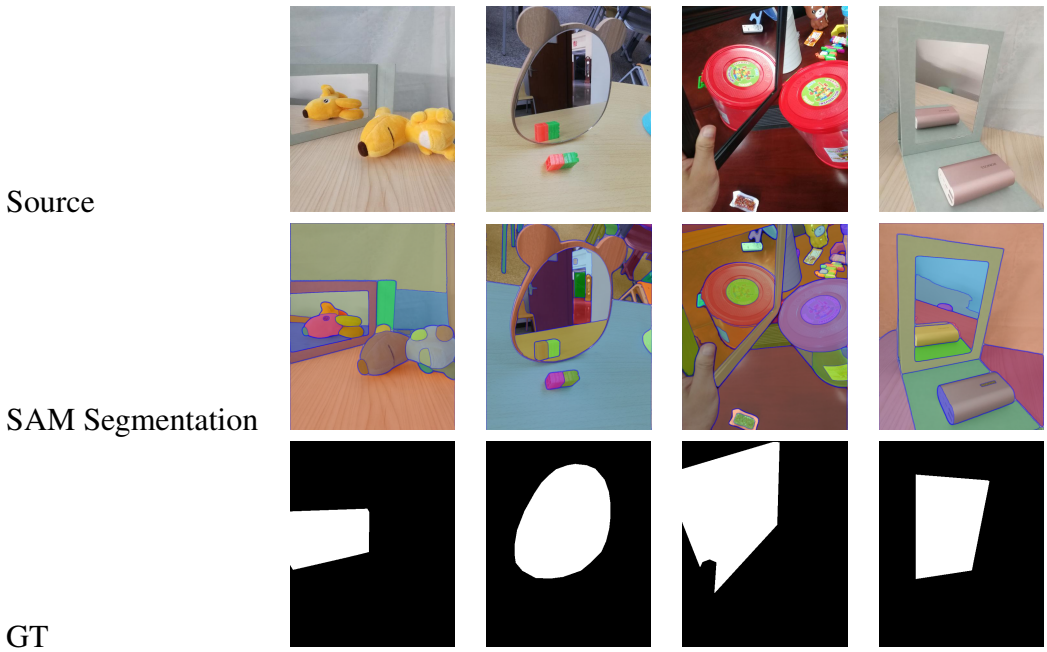


Figure 2.1: Occasions where SAM fails on the MSD dataset by segmenting the objects inside of the mirror as opposed to the mirror itself.

images, requiring 5 days on 256 A100 GPUs. Because of the large dataset, it has been found to outperform many previous purpose-built models.

SAM is unable to provide labels for the object categories for segmentation, instead it outputs all possible objects in the image. Therefore to test SAM's capability for mirror detection, the Intersection over Union (IoU) between the predicted region and the ground truth is calculated for each region outputted by SAM, and the result with the highest IoU is selected to be the prediction for the mirror region.

The results for mirror detection using SAM are lower than on purpose-built models. The reason for this is due to previous points mentioned where instead of detecting the entire mirror, SAM (similar to other object detection models) predicts and segments the objects inside of the mirror. This can be seen in diagram 2.1.

The accuracy of SAM on mirror and glass regions has already been analysed. Han et al. (2023) tested SAM on the Mirror Segmentation Dataset (MSD) Yang et al. (2019) and Progressive Mirror Dataset (PMD) Lin et al. (2020). The results from Han et al. (2023) are given in tables 2.2 and 2.3. As shown in table 2.3 SAM performs comparable to MirrorNet on the PMD dataset. It should be noted that MirrorNet was the first specialised single-image model for mirrors, it is still used as the most common benchmark, but it is far from the SOTA now as discussed later. However, as shown in table 2.2, SAM's performance on the MSD benchmark is very unsatisfactory, as it is considerably worse than MirrorNet. The reason for the vast difference between SAM's performance on PMD and MSD datasets is likely because the PMD dataset has more images of mirrors captured from a distance and offers a clearer view of the boundaries of mirror regions compared to the MSD benchmark, which has more images captured from a close range. Since the MSD dataset is a close range, this leads SAM to be more

Method	IoU↑	Acc↑	F_β↑	MAE↓	BER↓
MirrorNet	78.95	0.935	0.857	0.065	6.39
SAM	51.57	0.876	0.817	0.124	23.17

Figure 2.2: Experimental comparison between MirrorNet Yang et al. (2019) and SAM Kirillov et al. (2023) on the MSD dataset Yang et al. (2019). Results are from Han et al. (2023).

Method	IoU↑	Acc↑	F_β↑	MAE↓
MirrorNet	62.50	96.27	0.778	0.041
SAM	64.75	94.75	0.861	0.0525

Figure 2.3: Experimental comparison between MirrorNet Yang et al. (2019) and SAM Kirillov et al. (2023) on the PMD dataset Lin et al. (2020). Results are from Han et al. (2023).

likely to segment objects inside the mirror than recognise the mirror itself.

2.3.2 Mirror Detection without Deep Learning

Before the popularisation of deep learning technologies, researchers explored various methods for mirror detection leveraging classical computer vision techniques. These approaches primarily focus on exploiting specific properties of mirrors, such as reflectivity, geometry, and the unique interactions between light and surfaces. Three core papers contributed to the development of mirror detection methods without relying on deep learning algorithms.

The first study, ”On Solving Mirror Reflection in LIDAR Sensing” Yang and Wang (2011), explores the challenges of detecting mirror surfaces using LIDAR technology. The research introduces a Bayesian framework to identify and track mirrors, leveraging the geometric property of mirror symmetry. By focusing on the spatiotemporal aspects of reflections, this work provides a robust solution for integrating mirror detection into occupancy grid maps and localisation frameworks for mobile robots, demonstrating the method’s efficacy with real-world data.

In the realm of RGBD (RGB + Depth) data, ”MatterPort3D: Learning from RGB-D Data in Indoor Environments” Chang et al. (2017) indirectly contributed to mirror detection. While the primary focus is on semantic scene understanding, the comprehensive dataset and developed methodologies offer insights into handling reflective surfaces, including mirrors, within indoor settings. This research underscores the importance of semantic segmentation and classification in enhancing visual perception systems, indirectly facilitating mirror detection by understanding their interaction with the surrounding environment.

The third study, ”Reconstructing Scenes with Mirror and Glass Surfaces” Whelan et al. (2018), directly addresses the complexity of accurately detecting and reconstructing scenes containing mirrors and glass surfaces. By utilising an AprilTag Olson (2011) attached, the researchers developed an automatic pipeline capable of distinguishing

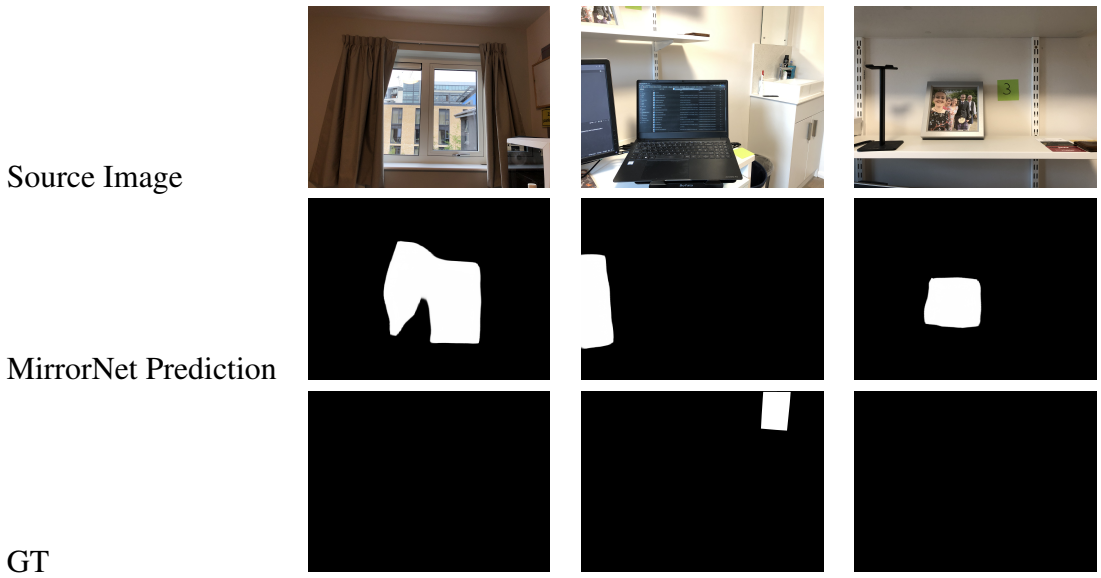


Figure 2.4: Occasions where the contextual contrasted feature extraction (CCFE) method fails.

between reflective surfaces and their surroundings. This method not only improves scene reconstruction accuracy but also enables realistic rendering of environments with reflective elements. The paper highlights the significance of integrating physical tags and exploiting reflective properties for mirror detection in 3D scanning applications.

These studies highlight mirror detection techniques' evolution in the pre-deep learning era. From leveraging LIDAR's geometric insights and RGB-D data for semantic scene understanding to employing physical tags for direct mirror and glass detection and reconstruction. There is no work on detecting mirrors with just RGB data without deep learning techniques. These foundational works paved the way for subsequent advancements in computer vision, offering valuable lessons on the nature of reflections.

2.3.3 Specialised Single Image Models

"Where is My Mirror" by Yang et al. (2019) was the first to address and solve the problem of mirror detection in still images using deep learning techniques. Their model - MirrorNet - took inspiration from how humans detect mirrors and reflections by focusing on the edges and looking for contrasting features between the contexts inside and outside mirror regions. The system takes a single image as input and extracts multi-level features using the feature extraction network (FEN). The deepest features that contain low-level semantics are then fed to the Contextual Contrasted Feature Extraction (CCFE) module to learn the contextual contrasted features for locating the mirrors. The issue with this method and similar methods is that contrasting features inside and outside of a frame appear in a lot more places than just mirrors such as photographs, windows, and computer monitors - this is illustrated in Figure 2.4.

This was improved upon in the paper "Progressive Mirror Detection" from Lin et al. (2020). The paper presents a novel approach for mirror detection, emphasising the

importance of understanding global scene semantics for the mirror detection problem. The authors observe that the content inside a mirror reflects its surrounding context and propose a new model - PMD-Net - that progressively learns the content similarity between the inside and outside of the mirror while explicitly detecting the mirror edges. The model works using two new modules - a Relational Contextual Contrasted Local (RCCL) module that extracts and compares mirror features with their corresponding context features, and an Edge Detection and Fusion (EDF) module that learns the features of mirror edges in complex scenes via explicit supervision. The proposed model demonstrated superior performance compared to MirrorNet, however, was not without issues. Firstly, because of the reliance on the EDF module, it often fails on reflective surfaces without a hard boundary or frame. Secondly, although the RCCL module is much better than the CCFE from MirrorNet, it is still far from perfect and fails often, especially in more complex scenes.

Following PMD-Net, the landscape of mirror detection witnessed several advancements. One notable development was SANet Huang et al. (2023), leveraging semantic associations as a heuristic grounded in the observation that mirrors frequently accompany commonplace objects like sinks or dressing tables. Despite surpassing PMD-Net's performance, SANet encounters limitations where mirrors are positioned in unconventional locations, such as ceilings or outdoor settings. In a different approach, Mirror-YOLO Li et al. (2022) adapts the methodology of the YOLOv4 architecture Bochkovskiy et al. (2020) to tackle mirror detection. While this adaption does not match the performance of models explicitly designed for mirror detection, it brings a significant advantage in efficiency, courtesy of its foundation in the YOLOv4 framework. Meanwhile, VCNet Tan et al. (2023) employs chirality (the concept that reflections in mirrors are inversions of their surroundings) as its core mechanism for identifying mirrors. This model, however, struggles when mirrors are partially hidden behind other objects. Furthermore, certain models exploit RGBD data Tan et al. (2021); Mei et al. (2021) integrating depth information with standard RGB imagery. This necessitates specialised hardware and lags behind state-of-the-art models.

The current SOTA model for the mirror detection in images problem is SATNet Huang et al. (2023). A dual-path symmetry-aware transformer-based mirror detection network that leverages the loose symmetry relationship between a real object and its reflection in a mirror. The authors observe that real-world objects and their reflections in mirrors maintain semantic or luminance consistency, even though they may not be strictly symmetric in position or orientation. SATNet includes two novel modules: Symmetry-Aware Attention Module (SAAM) and Contrast and Fusion Decoder Module (CFDM). SAAM captures symmetry relations by utilising a transformer backbone to model aggregate global information in images, extracting multi-scale features in two paths. CFDM fuses the dual-path features and refines prediction maps progressively to obtain the final mirror mask. Experimental results demonstrate that SATNet outperforms both RGB and RGBD mirror detection methods on all available mirror detection datasets.

Although not SOTA another model worth mentioning is HetNet He et al. (2023). HetNet is designed to initially detect potential mirror regions through low-level understandings and then finalise predictions by combining high-level understandings. The network employs a multi-orientation intensity-based contrasted module (MIC) and a reflec-

tion semantic logical module (RSL) to predict potential mirror regions by low-level understanding and analyse semantic logic in scenarios by high-level understandings, respectively. HetNet surpasses all other performance methods, with the sole exception of SATNet, while also maintaining a substantial edge in computational efficiency compared to SATNet.

2.3.4 Specialised Video Based Models: VMD-Net

The paper "Learning to Detect Mirrors from Videos via Dual Correspondences" Lin et al. (2023) is the first and only to address the challenge of video mirror detection (VMD) using deep learning methodologies. The authors propose a novel approach called VMD-Net. This approach capitalises on both intra-frame (spatial) and inter-frame (temporal) correspondences to detect mirrors with high accuracy. The dual correspondences approach uses two adjacent frames and another randomly sampled frame from the video as input to the model. This lets the model note correspondences that are not consistently present across all frames.

A significant contribution of this work is the introduction of the first large-scale dataset for video mirror detection, named the video mirror detection dataset (VMDD), comprising of 14,987 frames from 269 videos, each accompanied by manually annotated masks. This dataset is designed to facilitate research and development in the field by providing a diverse and comprehensive collection of scenes for training and evaluation. This dataset is discussed more in chapter 3.

VMD-Net is distinguished by its dual correspondence (DC) module, which consists of two stages. The first stage of the DC module is to learn the intra-frame correspondences by looking at the short-term correspondences between the two adjacent frames of the input. The second stage of the dual correspondence module focuses on long-term inter-frame correspondences by using the randomly sampled third input frame. This enables VMD-Net to learn both inter-frame and intra-frame correspondences at differing temporal and spatial scales.

The DC module can be seen in figure 2.6. To start, the three input images I_t, I_{t+1}, and I_n (for the two consecutive frames, and a third randomly selected frame respectively) are split into their low-level features and high-level features - F_{low} and F_{high}. Unlike image-based mirror detection models which use features at all stages (from 1^{st} to the 5^{th} scales), VMD-Net only uses the low-level features at the 2^{nd} level (F_{low}) and the high-level features at the 5^{th} scale (F_{high}).

Stage 1 of the DC module takes both the low-level features and the high-level features as input. First, a GR block from Lin et al. (2020) Progressive Mirror Detection is used for each high-level feature to extract the intra-frame correspondences. The GR block can efficiently and effectively extract the correspondences between contents inside and outside of mirrors by modeling spatial corresponding relations in a single image. For the randomly sampled third frame, the high-level correspondence-aware features are directly concatenated with the corresponding low-level features in the decoder. This outputs an intermediate map P_n. The high-level correspondence-aware features extracted from the two adjacent frames are forwarded to a cross-attention module

Method	IoU↑	Acc↑	F_β↑	MAE↓
MirrorNet	0.505	0.855	0.681	0.145
PMDNet	0.532	0.872	0.749	0.128
VCNet	0.539	0.877	0.749	0.123
VMD-Net	0.567	0.895	0.787	0.105

Figure 2.5: Experimental comparison between MirrorNet Yang et al. (2019), PMDNet Lin et al. (2020), VCNet Tan et al. (2023), and VMD-Net Lin et al. (2023). All models were trained on the VMDD-Test dataset and tested on the VMDD-Test set. Results are from Lin et al. (2023).

from Huang et al. (2019) to learn the short-term temporal correspondences. These temporal correspondences (C_{high} and C_{low}) are forwarded to another decoder to obtain intermediate predictions P.

Stage 2 of the DC module takes all the high-level features (F^t_{high}, F^{t+1}_{high}, and F^n_{high}) and intermediate maps (P_t, P_{t+1}, and P_n) as inputs. These are forwarded to a reverse cross-attention module. This module aims at explicitly exploiting the correspondence between the contents inside and outside of the mirrors in different frames for long-range temporal correspondences. It explicitly models correspondence between I_t and I_n. This is done by multiplying the high-level features with their corresponding intermediate prediction map and normalising using a sigmoid function. P is then reversed to obtain the reversed prediction map, that indicates the potential non-mirror regions. Since mirrors flip real-world objects horizontally, a horizontal flip to the input non-mirror features is conducted to model the potential relation of mirror reflections. The inter-frame correspondences are then extracted using the cross-attention module.

Experimental evaluations showcase VMD-Net's superior performance over state-of-the-art image-based approaches. The results from the Learning to Detect Mirrors from Videos via Dual Correspondences Lin et al. (2023) paper are given in table 2.5.

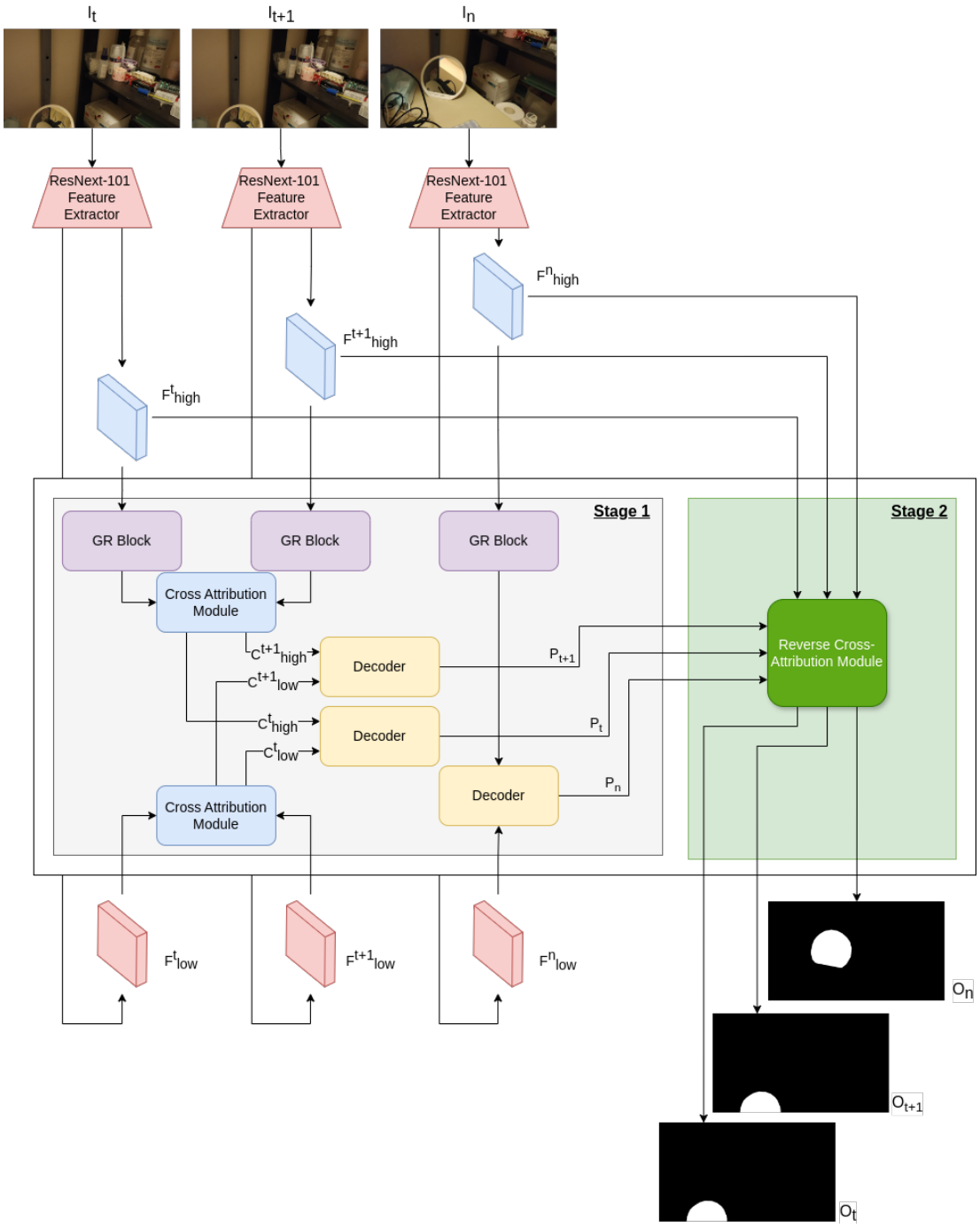


Figure 2.6: The framework of the VMD-Net model. I_t, I_{t+1}, and I_n are the input frames. Specifically, I_t and I_{t+1} are two adjacent frames while I_n is a randomly selected frame from the same video. O_t, O_{t+1}, and O_n are the final output maps for the corresponding input frames.

Chapter 3

Dataset

For our project, three separate datasets were used. Firstly the VMD Dataset (VMDD), first created for VMD-Net Lin et al. (2023), is an annotated dataset containing a total of 14,987 image frames across 269 videos. This dataset is split into a training set (VMDD-Train) of 7,835 frames and a test set (VMDD-Test) of 7,152 frames. Secondly, a relatively large unlabeled dataset containing 219,053 unique image frames across 525 videos was created for semi-supervised approaches. Finally, a very small annotated dataset containing just 984 video frames across 8 videos was created to measure the adaptability of the model to dissimilar data (since there is a high degree of similarity between VMDD-Test and VMDD-Train).

3.1 Existing Dataset

The paper ”Mirror Detection in Videos via Dual Correspondence” from Lin et al. (2023) created an annotated dataset for mirror detection in videos named Video Mirror Detection Dataset (VMDD). VMDD consists of 269 videos and a total of 14,987 individual frames. The frame rate is 30 frames per second and each frame has a resolution of 1280x720 pixels.

However, on inspection of this dataset, it can be seen that all videos share a lot of similarities. They all take place in one of two environments - either a household or inside of a furniture store; they all have the mirror in focus and in the foreground; and they all have the mirror in frame for the entire duration of the video. This can be seen in figure 3.1. Each of these images in the figure is from the first frame of different videos in the training and test set in VMDD. As can be seen, there is a high degree of similarity between images inside the training and test set. This high degree of similarity is quantified below in section 3.4. Noting this similarity was the pivotal moment in our project as it allowed us to hypothesise that such a similar dataset would lead to adaptability issues - which could then be remedied by using semi-supervised learning techniques.

The VMD dataset has 14,987 individual frames with a ground truth mask for each one. VMD is larger compared to the previous MSD and PMD datasets which have 4,018

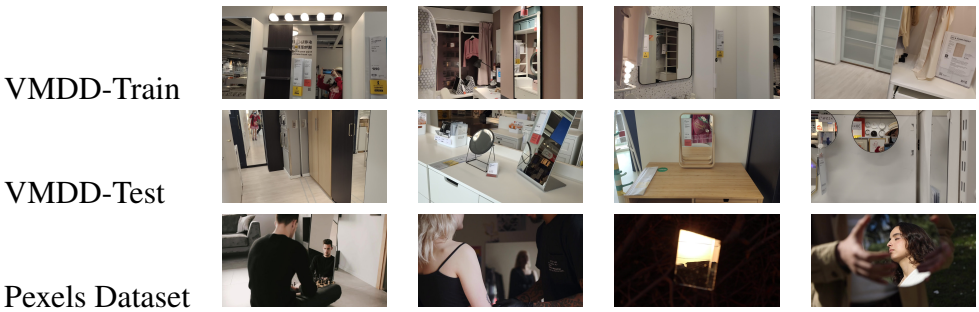


Figure 3.1: Visual representation of the similarity between the different datasets. Apparent high similarity in the VMDD-Train and VMDD-Test datasets, and less similarity in the Pexels dataset.

Yang et al. (2019) and 6,461 Lin et al. (2020) supervised frames respectively. However, when compared to larger computer vision datasets such as ImageNet Deng et al. (2009) which has about 1.2 million images for training and 50,000 for testing. Similarly, COCO Coc - a large-scale object detection and segmentation dataset, has 328,000 annotated images. As mentioned before, even more recently, Meta AI's Segment Anything Model Kirillov et al. (2023), was trained on 11 million images with a total of 1.1 billion segmentation masks. So, 14,987 images in VMDD means it is a comparatively small dataset to current standards. This is another reason why we hypothesised that the model's performance would drastically improve with access to a much larger unlabeled dataset through semi-supervised techniques.

3.2 Unannotated Dataset Created

Because the existing VMDD dataset is comparatively small and has a lot of similarities, we created a large unannotated dataset for semi-supervised approaches to overcome the issues of limited data size and high similarity between the data.

The dataset we gathered is publicly accessible with a royalty-free license under Creative Commons Zero license CCZ; Pex (b). The dataset was obtained from Pexels.com through their API Pex (a). Pexels.com is a free stock photo and video website where photographs and videos can be used for all personal, educational, research, and business purposes.

To create the dataset, we searched the Pexels API with the search term "Mirror". The quality for all videos downloaded was also filtered to be 1280x720 and were all in landscape mode at 30fps - this is so they are identical to the quality and frame rate used in the VMD dataset. This resulted in over 1000 unique videos, however, to make the dataset more manageable, only the first 600 were used. These 600 videos were then manually verified by two separate observers to ensure that they contained a visible mirror and didn't contain any video editing (e.g. black and white). An example of the types of videos removed can be seen in figure 3.2 Out of the 600 videos, after verification, we were left with 533 unlabelled videos. We randomly choose a subset of 8 of these and choose to manually annotate these as described below. The 8 we annotated

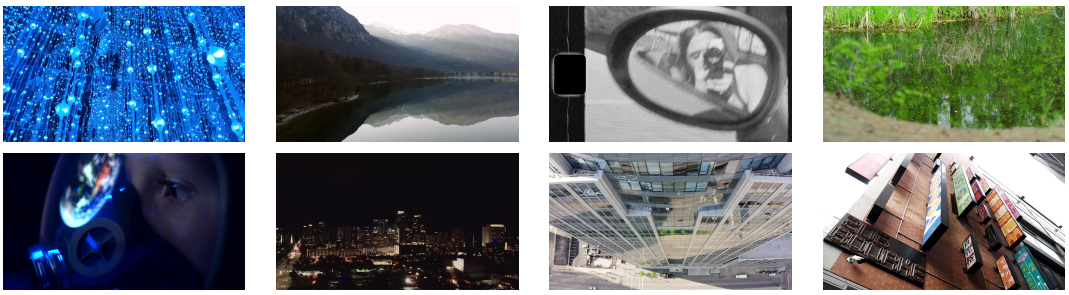


Figure 3.2: Examples of videos that were manually removed from the dataset. Some contain reflections from non-mirrors (e.g. astronaut helmet, water surface), some contain mirrors but are too stylistic (black and white image), and some don't contain any mirrors (such as the city skyline, and the signs).

were removed from the dataset used for semi-supervised learning to avoid any issues related to training on the test set.

3.3 Annotated Dataset Created

Because of the high degree of similarity in the training and test set mentioned before, we doubted how well the model would adapt to videos that contain mirrors but are less similar to the data used to train and test the model. Hence a small new annotated dataset was created consisting of videos that are dissimilar to each other and the VMD dataset. Consisting of videos from different environments. Due to the time constraints of the project, the scope of creating an annotated dataset was limited so this dataset only consists of 984 annotated video frames.

This dataset was created by randomly choosing 8 videos from the unlabeled dataset created and then manually annotating them. To annotate the data we used V7Labs V7L. This allowed us to create a pixel-level mask for each frame of the video.

3.4 Dataset Analysis

To quantify the similarity of the videos in each respective dataset, we used cosine similarity. Little work has been done on the similarity between whole videos so we measured the similarity between the first frame of each video - although this is not a perfect metric, it is a lot less computationally expensive than measuring the similarity between each frame and gives us a good approximation of the similarities of the whole video.

The first frame of each video was compared to the first frame of every other video. To calculate the cosine similarity of images, the first step is to use representation learning to learn a representation from the original image. This enables us to obtain a low-dimensional feature vector that is a good estimation of the relevant features of a given image frame.

After we have the vector embedding of the video frame, we can measure the similarity

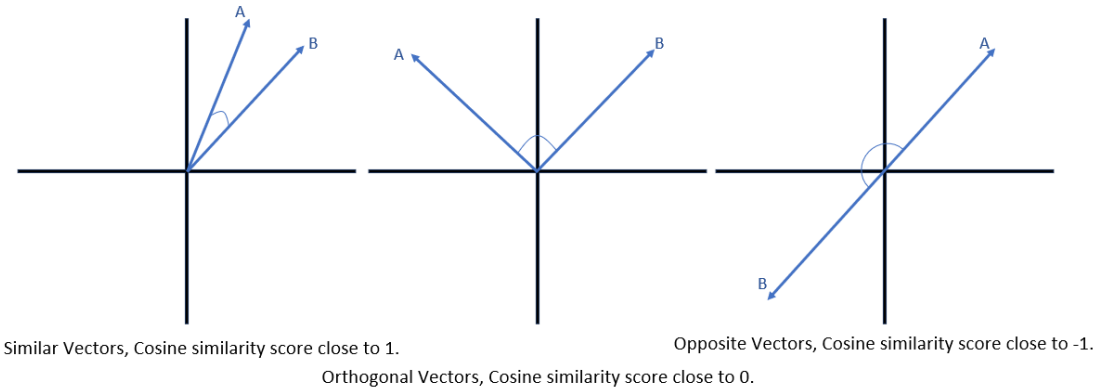


Figure 3.3: Vector diagram example showing vectors of different similarities. Similar vectors will have a score close to 1, opposite vectors will have a score close to -1.

between two video frames by using the cosine similarity. Given two vectors A and B, when projected into a feature space, vectors A and B will lie close to each other if they are similar and the distance between them is small, otherwise, they will lie far apart. A diagram showing vector similarities is given in figure 3.3. The formula for cosine similarity is given in formula 3.1. The cosine similarity will range from -1 to 1. We then scale this similarity so it is in the range 0 to 1 (0 is dissimilar, 1 is similar) so it is more interpretable.

$$Cosine(A,B) = \frac{A \cdot B}{|A||B|} \tag{3.1}$$

We perform the cosine similarity for two different use cases. Firstly intra-dataset similarity is the similarity of videos within one dataset by measuring the mean similarity between each video in the dataset and every other video in the same dataset. Secondly, inter-dataset similarity, which is the similarity between the two datasets by measuring the mean similarity between each video in one dataset and all the videos in the other dataset.

The results for the intra-dataset similarity are given in figure 3.6 and the results for inter-dataset similarity are given in figure 3.7. These results line up with our manual observation from figure 3.1. There is a high level of similarity of videos in the existing VMDD-Test and VMDD-Train datasets, while there is less similarity between the new Pexels dataset. The distribution of cosine similarities between different videos is given in 3.8. This quantification, confirms there is indeed an issue of data similarity in the existing VMD dataset, and since the newly created Pexels unlabelled dataset is more diverse, models trained using semi-supervised approaches on this new dataset should be more adaptable to less similar data.

We also analysed the distribution of video lengths in the dataset, the results can be seen in figure 3.4. As can be seen, the VMDD faces another major issue with the majority of videos in the dataset being exactly 60 frames long. This is because, during the dataset

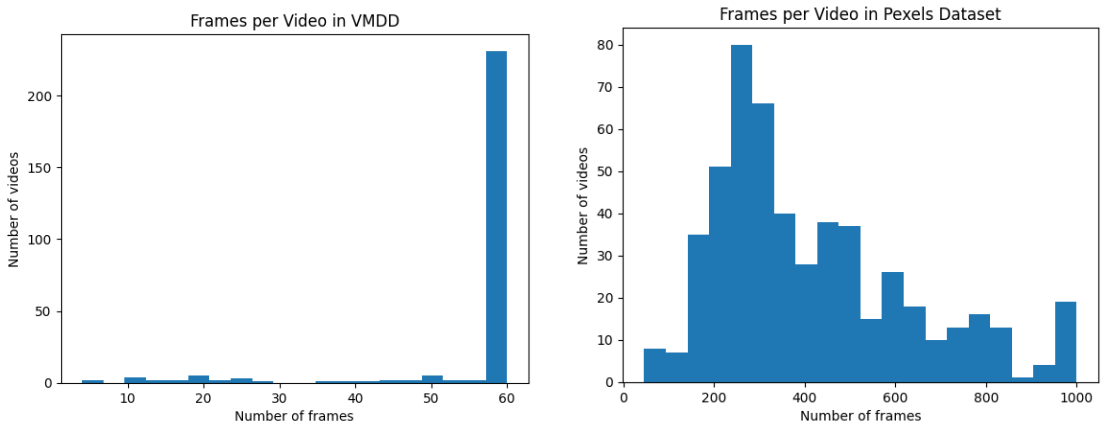


Figure 3.4: Distribution of dataset lengths as measured in the number of frames. As can be seen, VMDD is highly concentrated at 60 frames, whereas the new Pexels dataset is a lot more distributed around longer videos.

	Pexels Dataset	**VMDD Dataset**
Number of Videos	525	269
Average Duration (Frames)	417.24	55.93
Standard Deviation in Duration (Frames)	218.24	11.93
Total Duration (Frames)	219,053	14,987

Figure 3.5: Statistical analysis of the created dataset and the existing VMDD dataset. All videos are in 30fps to convert frames to seconds.

creation, longer videos were cropped into multiple shorter videos of 60 frames each to create multiple videos. This causes an issue as the VMD-Net DC module chooses a random third frame from the chosen video, this works well for short videos where the random third frame will be fairly close to the other two frames, however, in longer videos this may create difficulties - leading to further decreased adaptability.

Unfortunately, we couldn't analyse the distribution of mirror areas in our dataset as our dataset has no mask and so we couldn't calculate this.

Further relevant statistics of our dataset compared to the existing VMD dataset can be seen in table 3.5.

Dataset	**Mean Cosine Similarity**
VMD-Test	0.400
VMD-Train	0.431
Pexels Unlabeled	0.198
Pexels Annotated	0.276

Figure 3.6: Intra-dataset cosine similarity. The mean similarity between the first frame of each video in the dataset and every other video within the same dataset.

Dataset	Mean Cosine Similarity
VMD-Test & VMD-Train	0.388
VMD-Test & Unlabeled	0.163
VMD-Train & Unlabeled	0.160

Figure 3.7: Inter-dataset cosine similarity. The mean similarity between the first frame of each video in the dataset and all videos in the other dataset.

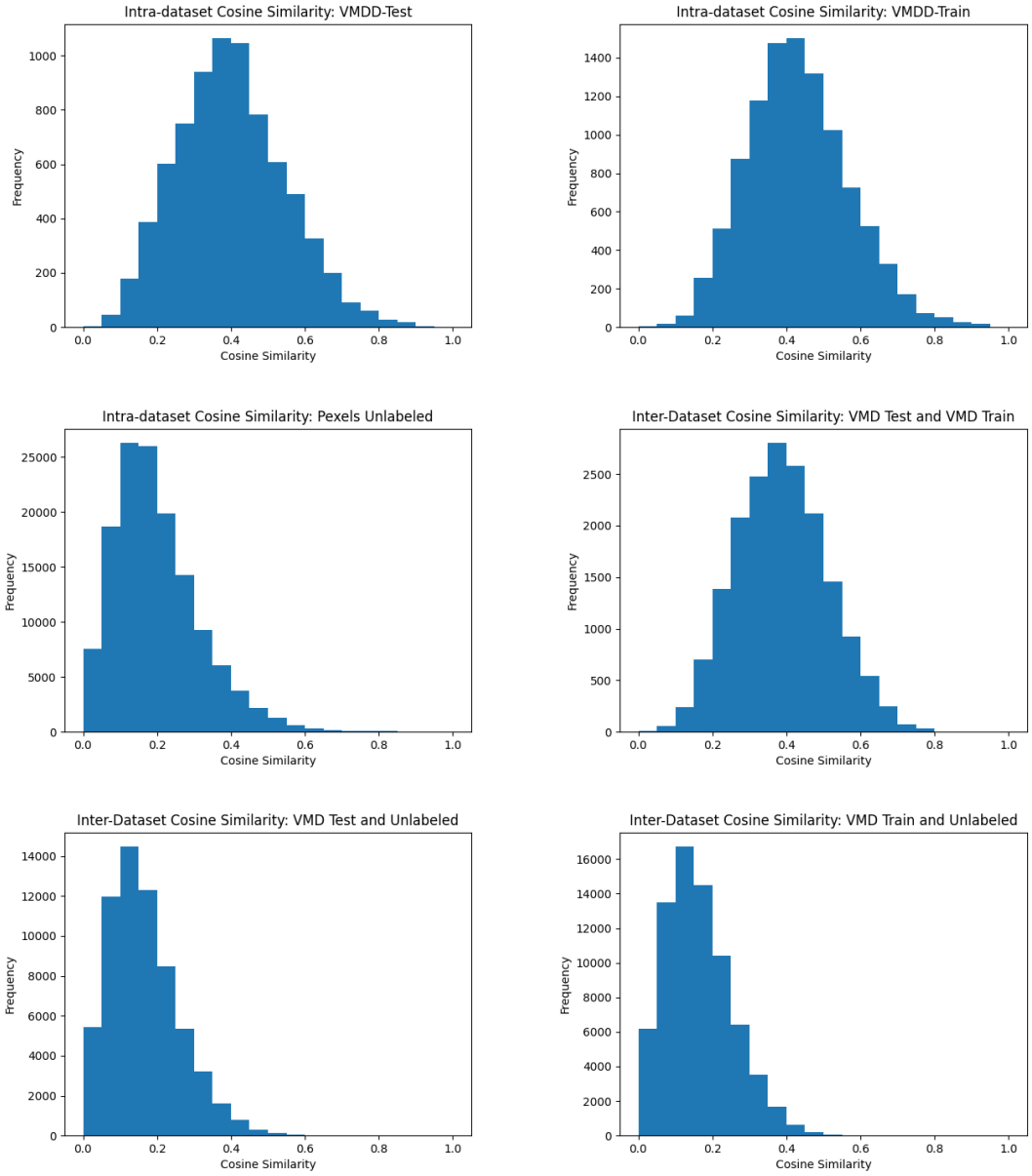


Figure 3.8: Histograms showing the distribution of the cosine similarities between the videos in different datasets. As can be seen, there is a much higher similarity between videos in the VMDD-Test and VMDD-Train datasets when compared to the Pexels datasets.

3.5 Limitations of the Dataset

There are two main issues with the datasets we created. Firstly, the annotated dataset is extremely small, consisting of just 8 videos and 984 video frames. Such a small dataset being used for analysis may provide inaccurate results that don't match real-world performance. However, since this is only meant to be used alongside the existing VMDD-Test set, to show the adaptability of the model to more diverse data (data that isn't similar to data in the training and test set), this should be satisfactory. However, if we had more time, creating a much larger annotated dataset would be beneficial to provide more accurate results.

The other issue is that the dataset was collected from stock videos, and so the quality of the videos is professional-grade. As discussed earlier, the real-world applications of mirror detection usually revolve around robotic navigation or surveillance where the video quality will be substantially less. This isn't an issue exclusive to our dataset though and is present in all mirror datasets (MSD, PMD, and VMDD). This is because the models created are more to show theoretical approaches that could be modified for real-world applications with the correct dataset, as opposed to creating models for a specific real-world situation - which might be more difficult to modify to another domain.

Finally, as discussed briefly, there is also an issue with the way we calculated the similarity between the various datasets. Right now, to measure the similarity, we vectorise only the first frame and use the cosine similarity between the first frames of the video. This is because, to our knowledge, there are no models for vectorising videos concerning calculating the cosine similarity between them. This has been proposed as a future project in the future work section. Another approach would be to calculate the similarity between each frame in the video and each frame in the other video. However, this would grow combinatorially with the length of videos and so would quickly become too computationally expensive to be feasible.

Chapter 4

Method

4.1 Evaluation Metrics

For evaluating our model, we use fairly standard metrics designed for binary segmentation tasks. Specifically, we will be using Intersection over Union (IoU), Mean Absolute Error (MAE), Pixel Accuracy, and Balanced Error Rate (BER). These metrics provide a comprehensive assessment of our model's performance from various perspectives. Below, we refine the descriptions of these metrics to clarify what each specifically measures in our use case.

Pixel Accuracy quantifies the proportion of pixels in the segmentation mask that are correctly classified, considering both mirror and non-mirror regions. It is calculated as the sum of true positives and true negatives divided by the total number of pixels. This metric offers a straightforward measure of overall correctness but may not fully capture the model's performance in imbalanced datasets where one class significantly outnumbers the other. The formula for pixel accuracy is detailed in 4.5.

F1 Score is the harmonic mean of precision (the proportion of true positive predictions in all positive predictions) and recall (the proportion of true positive predictions in all actual positives), providing a balanced measure of the model's accuracy concerning both mirror and non-mirror regions. It addresses the limitations of pixel accuracy in imbalanced datasets by weighting false positives and false negatives equally, making it a more reliable indicator of model performance when the number of non-mirror regions exceeds that of mirror regions. The equation for the F1 score is given in 4.3.

Intersecion over Union assesses the model's ability to accurately delineate mirror regions by calculating the ratio of the area of overlap between the predicted segmentation mask and the ground truth mask to the area of their union. A high IoU score indicates precise segmentation of mirrors, effectively distinguishing them from non-mirror regions. The formula for IoU is provided in 4.2.

Balanced Error Rate computes the average error across both classes (mirror and non-mirror) independently, thereby mitigating the bias introduced by class imbalance. It evaluates the model's ability to correctly identify mirror regions while also avoiding false identifications in non-mirror areas. This metric is particularly useful for highlighting

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - y_i| \tag{4.1}$$

Figure 4.1: Mean absolute error evaluation metric formula. x_i is the predicted value, y_i is the ground truth value. n is the sample size.

$$IoU = \frac{TP}{TP+FP+FN}(4.2) \quad F1 = \frac{2 \cdot TP}{2 \cdot TP+FP+FN}(4.3)$$

$$BER = \frac{1}{2}\dot{(}\frac{FN}{TP+FN} + \frac{FP}{TN+FP})(4.4) \quad Accuracy = \frac{TP+TN}{TP+TN+FP+FN}(4.5)$$

Figure 4.2: Equations for evaluation metrics used. TP is the count of true positive pixels. FP is the count of false positive pixels. TN is the count of true negatives. FN is the count of false negative pixels.

deficiencies in detecting the less prevalent class. The equation for BER is outlined in 4.4.

Mean Absolute Error measures the average magnitude of errors in the pixel values of the segmentation mask. It calculates the absolute difference between each pixel in the predicted mask and the ground truth, averaged over all pixels. MAE provides an insight into the average error per pixel, offering a direct measure of how closely the predicted mirror regions align with the actual mirror regions. This metric is critical for understanding the model's precision at the pixel level, especially in applications requiring high fidelity in mirror detection. The equation for MAE is discussed in 4.1.

4.2 Experimental Setup

For our self-training and expectation maximisation approaches, our model was trained on a single high-performance NVIDIA A100 GPU with 80GB of memory - courtesy of Edinburgh Compute and Data Facility (ECDF). The maximum epoch was set to 10 and the semi-supervised start epoch set to 5. These hyperparameters could be tuned further, however, since we wanted to test different confidence thresholds, we decided to keep these constant between experiments so valid comparisons could be made. The semi-supervised confidence thresholds are tuned to find the optimal value for each semi-supervised approach - their value changes between experiments and will be given for each result. The batch size - which determines the number of samples processed before the model is updated - was configured to 5. The data loader workers - which indicates the number of sub-processes to use for data loading - was set to zero. An Adam optimiser with a learning rate of 1e-3 and a weight decay of 5e-4 was used. Another important hyperparameter is the scale - this changes the resolution of the image frames during training. A higher scale will be closer to the resolution of the original image whilst a lower scale will be more pixelated. Therefore, a higher scale will lead to greater accuracy, however, requires drastically more computing power. We decided to use a scale of 416 (so images are resized to a scale of 416x416) as this gave a good balance between performance and accuracy. All hyperparameters and values are given in table 4.3

Hyperparameter	Ours	co-training	co-training - HetNet	VMD-Net
Batch Size	5	6	6	8
Data Loader Workers	0	0	3	unstated
Learning Rate	1e-3	1e-3	5e-3	1e-4
Weight Decay	5e-4	5e-4	5e-4	5e-4
Momentum	0.9	0.9	0.9	0.9
Scale	416	416	416	384
Maximum Epoch	10	10	150	15
Semi-Sup Start Epoch	5	5	75	N/A

Figure 4.3: Hyperparameters for our models. ”Ours” refers to the hyperparameters used for self-training and expectation maximisation. ”co-training” refers to the hyperparameters used for VMD-Net for the co-training technique. ”co-training - HetNet” refers to the hyperparameters used for the HetNet model in the co-training technique. VMD-Net is the hyperparameters stated in Lin et al. (2023) ”Learning to Detect Mirrors from Videos via Dual Correspondences” Paper.

For co-training, we aimed to keep the experimental setup as close as possible to the setup for our models discussed before - we will be using the same loss function and the same method for calculating the confidence of a prediction. Since co-training requires two models, it was trained on three NVIDIA A100 GPUs with a total of 240GB of memory. The hyperparameters for co-training are also given in table 4.3. It is worth noting, for co-training, we also had to modify the batch size for the video model from 5 to 6. This is because co-training method required three A100 and the batch size has to be divisible by the number of GPUs. This difference in experimental setup is unlikely to cause any difference in results.

4.2.1 Loss Function

In binary segmentation tasks, Binary Cross-Entropy (BCE) is often employed as the standard loss function. It calculates the log loss for every pixel by comparing the ground truth label (either 1 or 0) with the predicted probability p. The BCE formula is given below:

$$BCE = -(y\log(p) + (1-y)\log(1-p)) \tag{4.6}$$

Figure 4.4: Binary Cross-Entropy Loss. y is the ground truth - either 1 or 0, p is the predicted probability.

However, a novel approach to optimising segmentation models is the Lovasz-Softmax loss function first introduced by Berman et al. (2018). The Lovasz function focuses on directly improving the IoU metric, surpassing the capabilities of the standard BCE in terms of segmentation performance.

The Lovasz-Softmax loss function is especially advantageous for models like VMD-Net, which aim to enhance segmentation accuracy. This function is derived from the concept

$$L = \sum_{i \in \{t, t+1, n\}} (L_h(P_i, G_i) + L_h(O_i, G_i)) \tag{4.7}$$

Figure 4.5: Lovasz loss function. t, t+1, and n refer to the three sample frames (two consecutive and third randomly chosen). P_i is the intermediate output, O_i is the final output, G_i is the ground-truth. These are explained more in diagram 2.6.

of the Lovasz extension of submodular functions, applied in the context of binary segmentation to directly target the optimisation of the IoU metric. The loss is calculated by considering the hinge loss (L_h) between the predicted segmentation mask and the ground truth. In our setup, both the intermediate and final outputs of the VMD-Net (P_i and O_i, respectively) are compared against the ground truth (G_i), as shown in figure 4.5.

$$L_h(\hat{Y}, Y) = max(0, 1 - (Y \cdot \hat{Y})) \tag{4.8}$$

Figure 4.6: Hinge loss function. $\hat{Y}$ is the predicted value, Y is the ground truth value.

4.2.2 Confidence Calculation

In the context of semi-supervised fine-tuning for pixel-level segmentation, calculating the confidence of model predictions on unlabeled video frames is a critical step. This process not only aids in assessing the reliability of the model's predictions but also plays a pivotal role in the self-training step, where only highly confident predictions are utilised to further train the model.

The confidence score for each video is calculated as follows:

1. **Segmentation Prediction:** For each frame in the video, the model generates a predicted segmentation mask. This mask indicates the likelihood of each pixel being part of a mirror region. This mask is unbounded (theoretically can go from -∞ to ∞), and is centered around 0 (<=0 is no mirror, >0 is mirror region). The confidence of a prediction is higher the further the prediction is away from 0. A prediction of -5 means the model is very confident the pixel isn't a mirror, a prediction of 0 the model is completely unsure, and a prediction of 1, the model is somewhat sure the pixel is a mirror.
2. **Confidence Score Calculation:** We then calculate the confidence in the prediction for each pixel. The confidence score for each pixel is calculated by applying a modified sigmoid function to normalise the output so the confidence falls in the range from 0 to 1. The function we use to calculate the confidence of a specific pixel segmentation prediction x is given below:

$$Confidence(x) = \left(\frac{1}{1 + e^{-|x|}} - 1 \right) \tag{4.9}$$

3. **Average Confidence:** The average confidence of a frame can then be calculated by dividing the sum of the pixel level confidence by the total number of pixels.

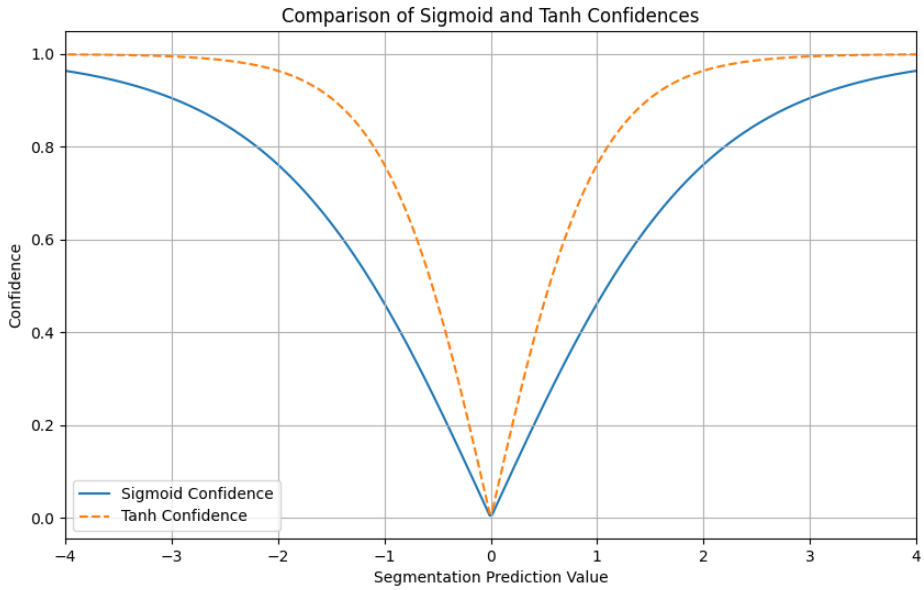


Figure 4.7: Plot showing the segmentation prediction value and corresponding confidence for both the sigmoid and tanh functions.

Likewise, The average confidence score for the entire video is obtained by dividing the total confidence by the number of frames processed. This average score represents the overall confidence in the model's predictions for the video.

4. **Confidence Threshold:** Videos with an average confidence score exceeding a predefined threshold are considered reliable. These high-confidence videos are then labeled accordingly and utilised in subsequent training cycles to refine the model's performance further.

As can be seen from figure 4.7, we could have also chosen to use the hyperbolic tangent function (tanh) to convert the segmentation prediction value into confidence. This would likely have some impact on the expectation maximisation method where the confidence of the prediction is used to weigh the loss function. However, we did not have enough time to redo all our tests with this modified confidence function.

4.3 Semi-Supervised Approaches

Semi-supervised learning sits in between supervised learning where all data is labeled and unsupervised learning where no data is labeled. It is particularly useful when acquiring a large set of labeled data is expensive or impractical, but unlabeled data is abundant. Semi-supervised learning leverages a small amount of labeled data along with a large amount of unlabeled data to improve learning accuracy. This project uses three key approaches within semi-supervised learning: self-training, expectation maximisation (EM), and co-training.

4.3.1 Self-training

self-training, also known as self-learning, is a simple yet effective semi-supervised technique. It begins with a small set of labeled data to train an initial model. The model is then used to predict labels for the unlabeled data. Predictions made with high confidence are added to the labeled set, and the model is retrained with this expanded dataset. This process iterates until a stopping criterion is met - this can be the desired validation loss or maximum epoch.

Previous research on self-training has demonstrated its efficacy in various domains. For example, when applied to word sense disambiguation, it showed significant improvement with each iteration of adding confidently predicted labels Yarowsky (1995). This approach has also been explored in the context of image recognition, where models incrementally improve as they ”teach themselves” from the unlabeled data pool Oliver et al. (2018).

The implementation details for self-training are given in diagram 4.8 and the pseudocode 1. First, the model is trained on the labeled data (VMDD-Train), then after 5 epochs, it labels the unlabelled Pexels dataset. The most confident predictions are then combined with the labeled data to give a combined dataset. The next epoch is then trained off of this combined dataset. Stages two and three are then repeated.

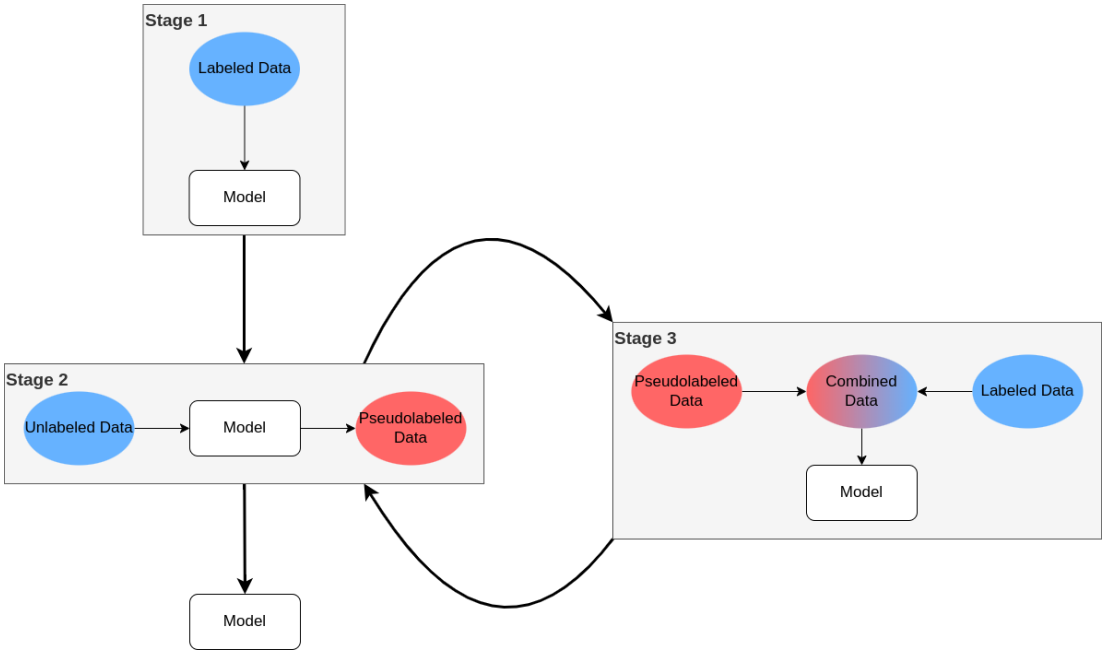


Figure 4.8: Diagram showing the key steps for the self-training and expectation maximisation approach. **Stage 1:** The model starts by training off of the labeled dataset. **Stage 2:** The model annotates the unlabelled dataset and assigns pseudo-labels. **Stage 3:** The most confident pseudo-labels are combined with the labeled data and the model is then trained off of this combined dataset. Stages 2 and 3 are then repeated until end of training.

Algorithm 1 pseudo-labeling self-training Algorithm

Initialise model M with labeled dataset $D_{labeled}$
Let $D_{unlabeled}$ be the dataset without labels
Define confidence threshold θ
repeat
    Predict labels for $D_{unlabeled}$ using model M to get $\hat{Y}$
    Select subset D_{pseudo} from $D_{unlabeled}$ where prediction confidence $\geq \theta$
    Augment $D_{labeled}$ with D_{pseudo} to get new training set D_{new}
    Retrain model M on D_{new}
until max epoch is met
return model M

4.3.2 Expectation Maximisation

Expectation maximisation (EM) is a more sophisticated approach that iteratively estimates the maximum likelihood estimates of parameters in statistical models, which depend on unobserved latent variables. In the context of semi-supervised learning, EM alternates between assigning (expectation step) the most likely labels to the unlabeled data based on current model parameters and then updating the model (maximisation step) to maximise the likelihood of the data given these labels.

Nigam et al. (2000) applied EM to text classification by using a small set of labeled documents and a large pool of unlabeled documents to improve classification accuracy. Their work highlights the EM algorithm's ability to effectively utilise unlabeled data by iteratively refining the model's understanding of the data distribution.

The implementation details for expectation maximisation are given in diagram 4.8 and pseudocode 2. The steps are fairly close to those used for self-training but with one key difference. In self-training the confident predictions on the unlabelled dataset are treated as ground truth values for the next training step. This is not the case, if the model is say 80% confident of classification, it should be treated as less reliable than a 90% confident prediction and the ground truth. This is why for the expectation maximisation we take the confidence of a classification into account during the maximisation step by considering the confidence scores as weights into the loss function for the model.

Algorithm 2 Expectation Maximisation Algorithm for Semi-Supervised Learning

1: Initialise model M with labeled dataset $D_{labeled}$
2: Let $D_{unlabeled}$ be the dataset without labels
3: Define confidence threshold θ
4: **repeat**
5: **E-step**: Estimate the most likely labels $\hat{Y}$ for $D_{unlabeled}$ using current model M
6: Calculate confidence scores for predictions on $D_{unlabeled}$
7: **Selection**: Select subset D_{pseudo} from $D_{unlabeled}$ with confidence scores $\geq \theta$
8: **M-step**: Retrain model M using both $D_{labeled}$ and D_{pseudo} considering the confidence scores as weights in the loss function
9: **until** convergence criteria are met or max epoch is reached
10: **return** model M

4.3.3 Co-training

Co-training is a multi-view learning approach that assumes the data can be described by two independent feature sets, which are sufficient for learning the task at hand. The core idea is to train two classifiers separately on each view, and then each classifier labels unlabeled examples for the other to learn from. This approach relies on the assumption that the two views are conditionally independent given the class label and that each view is sufficient for classification.

Blum and Mitchell (2000) introduced co-training in the context of web page classification, where the text on the web page and the text in the hyperlinks pointing to the page served as two independent views. Their work demonstrated that co-training could significantly reduce the need for labeled examples by exploiting the redundancy between the views.

Subsequent research has extended co-training to various domains and explored modifications to the original assumptions. For instance, the co-training framework has been adapted to scenarios where multiple views are not explicitly available, by creating artificial views through feature splitting or the use of different learning algorithms. The use of different learning algorithms is the co-training approach that was used in this project.

As seen in figure 4.9 and the pseudocode in 3 the co-training technique requires two separate models so that each model can be trained off of the other model's annotations. For the second model, we decided to go for an image-based mirror detection model as many existing image-based models use features unexplored in the video-based model (such as SANet Guan et al. (2022) using semantic associations, or VCNet Tan et al. (2023) using the visual chirality). As discussed in chapter 2 the SOTA image-based model is Symmetry Aware Transformer based mirror detection (SATNet) Huang et al. (2023), closely followed by Multi-level heterogeneous learning Hetnet He et al. (2023). We decided to test co-training with the SOTA image-based approach - it would be interesting to test if weaker models performed better in co-training, however, due to time constraints testing every model for co-training wasn't feasible. Hence, we decided to just try the SOTA under the assumption that this would produce the best

results. However, when trying to implement the co-training technique with SATNet, we encountered serious issues with dependency issues (the core issue is the ECDF compute cluster uses CUDA 11.0, but SatNet requires CUDA 10.1). Hence, we decided to go for the 2^{nd} best image-based SOTA - HetNet.

Figure 4.9 and pseudocode 3 show the dual-model approach to co-training. Similar to self-training, the core idea behind this approach is based on using pseudo-labels from the unannotated dataset to train the next iteration of the model. The difference in co-training is that instead of just the single model annotating and training, there are two separate models. The pseudo-labels from one model are passed into the training set for the second model and vice versa. In practice, we start by training the HetNet and VMD-Net for 75 and 5 epochs respectively on just the original VMDD-Train set. Then HetNet and VMD-Net both annotate the unlabeled dataset. The most confident predictions from HetNet are then passed into the dataset for training VMD-Net on the next iterations. Likewise, the most confident predictions from VMD-Net are passed into the dataset for training HetNet. VMD-Net is then trained for another single iteration on this new training set, whilst HetNet is trained for 15 iterations on its new dataset. This step of annotating, then retraining on the combined dataset is then repeated until we reach the maximum epoch for each model.

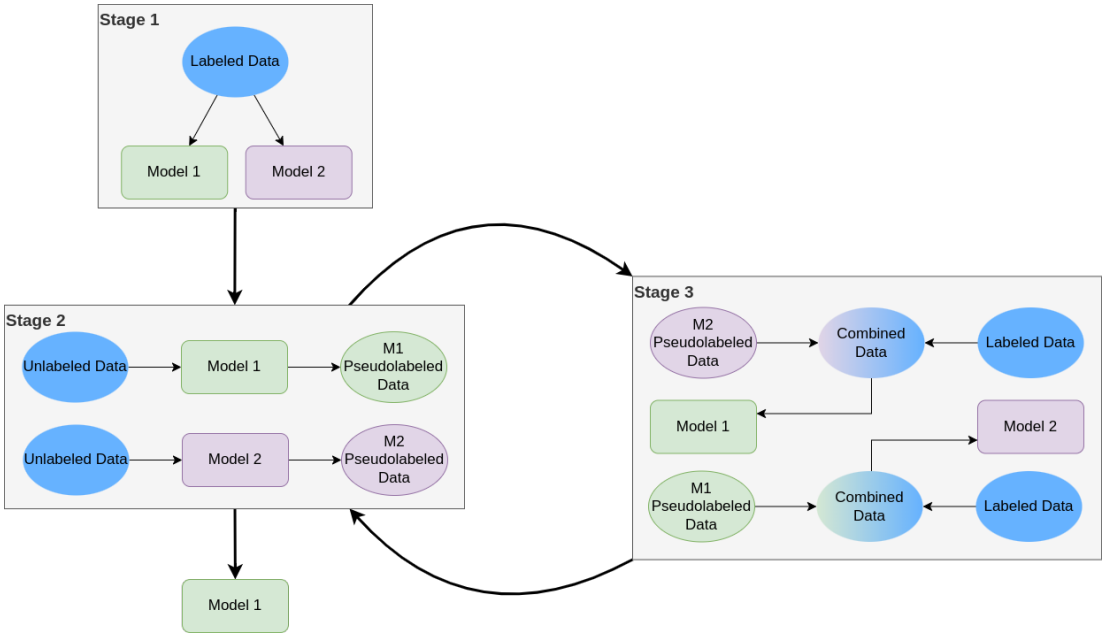


Figure 4.9: Diagram showing the key steps for the co-training technique. **Stage 1:** Both models start by training off of the labeled dataset. **Stage 2:** Each model annotates the unlabelled dataset and assigns pseudo-labels. **Stage 3:** The most confident pseudo-labels are combined with the labeled data. Model 1 is trained off of the pseudo-labels from model 2 and vice versa. Stages 2 and 3 are then repeated until end of training. In our case, Model 1 is the modified video model and model 2 is HetNet.

Algorithm 3 Co-training Algorithm

Initialise two models M_1 and M_2 with labeled dataset $D_{labeled}$
Partition $D_{labeled}$ into two views V_1 and V_2 respectively for M_1 and M_2
Let $D_{unlabeled}$ be the dataset without labels
repeat
    Train M_1 on V_1 and M_2 on V_2
    Use M_1 to predict labels on $D_{unlabeled}$ to create $D_{pseudo1}$
    Use M_2 to predict labels on $D_{unlabeled}$ to create $D_{pseudo2}$
    Select a confident subset from $D_{pseudo1}$ and add it to V_2
    Select a confident subset from $D_{pseudo2}$ and add it to V_1
    Optionally, update $D_{unlabeled}$ by removing the selected subsets
until stopping criteria is met
return models M_1 and M_2

Chapter 5

Results

5.1 Control Results

The results from the experiments can be seen in the following figures. The control results - testing VMD-Net with our experimental setup without any semi-supervised learning taking place can be seen in 5.1. A notable point of these results is that they are substantially lower than the results given in the original paper Lin et al. (2023) - the same results recorded in figure 2.5. This is due to our differing experimental setup. The different setups are shown in table 4.3. As can be seen, we have a significantly higher learning rate (1e-3 compared to 1e-4). The reason for this higher learning rate is that we were running multiple experiments and so we had to lower the maximum epoch from 15 to 10. Without lowering the maximum epoch, our experiments would have taken substantially longer to run. Because of decreasing the maximum epoch, we had to increase the learning rate to compensate. Because of this combination of higher learning rate and lower maximum epoch, our control results are lower than the original results stated. We did train the original VMD-Net with a higher maximum epoch and lower learning rate and were able to get results close to that achieved in the paper, however, due to the number of tests being run, having a large maximum epoch was infeasible for all experiments.

Test Set	IoU↑	Acc↑	F_β↑	MAE↓	BER↓
VMDD-Test	0.349	0.865	0.696	0.135	31.58
Pexels Labelled	0.382	0.803	0.686	0.197	31.07

Figure 5.1: Control Results: Model with parameters as specified in Experimental Setup section. Trained on VMDD-Train, tested on VMDD-Test and Pexels labeled.

5.2 Self-training

The self-training approach, a foundational semi-supervised technique, demonstrates notable improvements in mirror detection performance over control experiments, as evidenced by the data presented in figures 5.2, 5.3, 5.11, and 5.10. This method,

which relies on incrementally training the model with predictions that exceed a certain confidence threshold, illustrates a clear trend: as the minimum confidence threshold increases, so does the model's performance on the VMDD-Test set across most metrics. This trend holds true up to a critical point between 77.5% to 80% confidence levels for both intersection over union (IoU) and the balanced error rate (BER), where a slight deviation occurs. Notably, the performance on the VMDD-Test set surpasses that of the control results across all metrics, indicating the efficacy of self-training in enhancing model accuracy.

The results on the Pexels dataset provide additional insights into the adaptability of the self-training approach. Performance metrics improve consistently up to a confidence threshold of 75%-77.5%, where they peak, surpassing control results and showing comparable performance to the VMDD-Test set. This peak performance indicates the model's enhanced adaptability to diverse datasets, which may differ significantly from the data it was originally trained on. However, at the 80% confidence threshold, performance declines, likely due to the scarcity of videos exceeding this confidence level, which restricts the addition of new annotations in subsequent iterations and brings the model's performance closer to control levels.

This pattern of results underscores the potential of self-training as a strategy for improving mirror detection models, particularly when adjusting the confidence threshold for incorporating predictions into training. The initial increase in performance metrics with higher confidence thresholds suggests that carefully selecting predictions for retraining can lead to more accurate models. However, the observed decline at the very high thresholds highlights the balance needed between excluding less certain predictions and ensuring sufficient data for model refinement. These findings emphasise the importance of optimising confidence thresholds to maximise the benefits of self-training in diverse testing scenarios.

Min Confidence (%)	IoU↑	Acc↑	F_β↑	MAE↓	BER↓
65	0.331	0.861	0.684	0.139	32.61
70	0.332	0.861	0.685	0.138	32.62
75	0.344	0.864	0.697	0.136	31.65
77.5	**0.391**	**0.874**	*0.723*	*0.126*	**29.23**
80	*0.382*	*0.880*	**0.727**	**0.120**	*29.80*

Figure 5.2: Experimental Results from self-training approach, tested on VMDD-Test dataset. Best results in **Bold**, Second best in *Italics*.

Min Confidence (%)	IoU↑	Acc↑	F_β↑	MAE↓	BER↓
65	0.339	0.809	0.687	0.191	32.27
70	0.375	0.814	0.681	0.186	30.862
75	*0.408*	**0.830**	**0.735**	**0.170**	*28.98*
77.5	**0.440**	*0.822*	0.668	*0.178*	**28.82**
80	0.341	0.809	*0.693*	0.191	32.14

Figure 5.3: Experimental Results from self-training approach, tested on labeled Pexels dataset to measure adaptability. Best results in **Bold**, Second best in *Italics*.

5.3 Expectation Maximisation

The Expectation Maximisation (EM) approach to semi-supervised learning in mirror detection displays a performance trend similar to self-training, yet with notable differences. This technique, as illustrated in the experimental outcomes, shows an increase in performance metrics with rising confidence thresholds on the VMDD-Test dataset. Unlike the more consistent improvements observed in self-training, EM exhibits a pattern interspersed with fluctuations, suggesting a noisier progression in performance enhancements. This variability indicates that while the overall trend points towards better results with higher confidence levels, the journey there is less predictable.

Figures 5.4, 5.5, 5.10, and 5.11 reveal that the peak performance on the VMDD-Test set is achieved at an 80% confidence threshold, marking the highest recorded improvements across all metrics including IoU, Accuracy, F_β, MAE, and BER. This peak contrasts with the performance on the Pexels dataset, where adaptability peaks at a slightly lower confidence threshold before declining. Such a peak underscores the EM approach's nuanced response to varying datasets and the critical role of choosing an optimal confidence threshold for maximising model performance.

The EM method's performance on the Pexels dataset offers insights into the model's adaptability to disparate data. The best results are achieved at an 80% confidence threshold, similar to the VMDD-Test dataset, yet the path to this peak is characterised by notable variability. This adaptability, peaking before a drop at higher thresholds, suggests that an optimal balance in confidence levels is key to leveraging the EM approach effectively across different types of data.

Min Confidence (%)	IoU↑	Acc↑	F_β↑	MAE↓	BER↓
65	0.354	0.859	0.695	0.140	31.24
70	0.319	0.862	0.674	0.138	33.31
75	*0.362*	*0.873*	*0.708*	*0.126*	*31.03*
77.5	0.340	0.862	0.691	0.138	32.05
80	**0.402**	**0.875**	**0.729**	**0.125**	**28.74**

Figure 5.4: Experimental Results from Expectation Maximisation approach, tested on VMDD-Test dataset. Best results in **Bold**, Second best in *Italics*.

Min Confidence (%)	IoU↑	Acc↑	F_β↑	MAE↓	BER↓
65	0.339	0.809	0.690	0.191	32.54
70	0.354	0.813	0.710	0.187	31.59
75	0.385	0.808	0.701	0.192	31.17
77.5	*0.421*	*0.836*	**0.750**	*0.164*	*28.25*
80	**0.423**	**0.847**	*0.728*	**0.153**	**28.17**

Figure 5.5: Experimental Results from Expectation Maximisation approach, tested on labeled Pexels dataset to measure adaptability. Best results in **Bold**, Second best in *Italics*.

5.4 Co-training

The co-training approach, distinguished by its strategy of utilising two or more independent views of the data to mutually enhance learning, demonstrates notable success in the realm of mirror detection. Experimental results, as highlighted in figures 5.6, 5.7, 5.10, and 5.11, reveal a consistent pattern of improvement across various metrics as the minimum confidence threshold for incorporating predictions into the training set is increased. This pattern is observable in both the VMDD-Testset and the labeled Pexels dataset, with the optimal performance occurring at 80% confidence threshold for VMDD-Test and 77.5% for Pexels.

The incremental improvements seen with increasing confidence thresholds suggest that the co-training model becomes progressively better at discerning more challenging or ambiguous cases of mirror reflections as it becomes more selective in the data it learns from. This selectiveness, driven by high-confidence thresholds, ensures that only the most reliable predictions contribute to further learning, thereby enhancing the overall quality and reliability of the detection process.

Min Confidence (%)	IoU↑	Acc↑	F_β↑	MAE↓	BER↓
65	0.343	0.858	0.688	0.118	30.12
70	0.347	0.859	0.692	0.115	31.83
75	*0.376*	**0.882**	**0.721**	0.131	31.68
77.5	0.372	8.860	0.707	*0.104*	*29.36*
80	**0.384**	*0.866*	*0.719*	**0.102**	**29.35**

Figure 5.6: Experimental Results from Co-training approach, tested on VMDD-Test dataset. Best results in **Bold**, Second best in *Italics*.

Min Confidence (%)	IoU↑	Acc↑	F_β↑	MAE↓	BER↓
65	0.398	0.818	0.698	0.198	30.75
70	0.404	*0.824*	0.704	0.194	30.17
75	0.406	0.821	0.706	0.183	30.46
77.5	**0.421**	**0.831**	**0.716**	*0.181*	**28.83**
80	*0.417*	0.822	*0.712*	**0.162**	*29.85*

Figure 5.7: Experimental Results from Co-training approach, tested on labeled Pexels dataset to measure adaptability. Best results in **Bold**, Second best in *Italics*.

5.5 Summary

Across all semi-supervised learning models tested - self-training, Expectation Maximisation (EM), and Co-training - a consistent pattern emerges in their performance on the VMDD-Test and labeled Pexels dataset. Notably, all models exhibit superior accuracy, F_β, and MAE on the VMDD-Test set, suggesting a strong alignment with the characteristics of the dataset they were trained on. Conversely, for BER and IoU metrics, the models tend to demonstrate better performance on the Pexels dataset. This observation contradicts our initial hypothesis, which anticipated reduced performance across **all** metrics. A contributing factor to this discrepancy could be the limited sample size of the annotated Pexels dataset, previously identified as a potential limitation in chapter 3.

When comparing the peak results from each approach - in figures 5.8 and 5.9 - the Expectation Maximisation (EM) method stands out for achieving the highest overall performance on both the VMDD-Test set and the Pexels dataset. The EM method's success over the control model across multiple metrics not only validates the benefits of semi-supervised learning but also supports the hypothesis that utilising unlabeled data can indeed mitigate data similarity challenges, thereby boosting model performance.

The trend of performance peaking at certain confidence thresholds before a subsequent decline, particularly observed in the Pexels dataset, highlights a critical balance in selecting predictions for model retraining. This phenomenon likely results from the diminishing returns of adding high-confidence predictions beyond a certain point, where the model becomes overly constrained by the scarcity of qualifying data, reverting to performance levels akin to the control model.

In summary, the experimental results from the three semi-supervised approaches support the original hypothesis that using these semi-supervised approaches with vast unlabeled data can help to overcome issues of data similarity. All semi-supervised models outperform the control results - slightly on the VMDD-Test set and substantially on the Pexels dataset. Meaning the use of semi-supervised approaches allows the model to adapt better to dissimilar data while also performing better on the original test set.

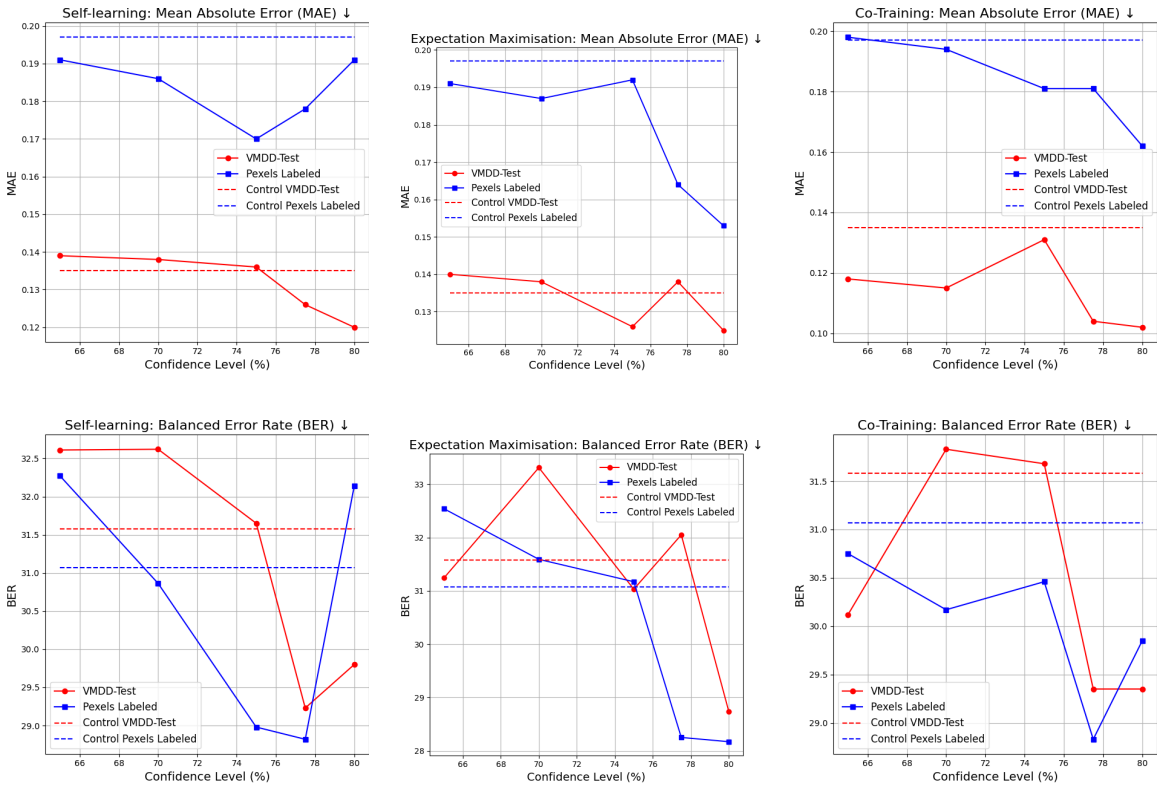


Figure 5.10: Line charts showing the Mean Absolute Error and Balanced Error Rate for each of the three semi-supervised approaches. The left column is self-training; the middle column is expectation maximisation; the right column is co-training.

Model	Min Confidence (%)	IoU↑	Acc↑	F_β↑	MAE↓	BER↓
Control	N/A	0.349	0.865	0.696	0.135	31.58
self-training	77.5	*0.391*	*0.874*	*0.723*	0.126	*29.23*
EM	80	**0.402**	**0.875**	**0.729**	*0.125*	**28.74**
Co-training	80	0.384	0.866	0.719	**0.102**	29.35

Figure 5.8: Best model from each approach on VMDD-Test dataset. Best results in **Bold**, Second best in *Italics*.

Model	Min Confidence (%)	IoU↑	Acc↑	F_β↑	MAE↓	BER↓
Control	N/A	0.382	0.803	0.686	0.197	31.07
self-training	75	0.408	0.830	**0.735**	*0.170*	28.98
EM	80	**0.423**	**0.847**	*0.728*	**0.153**	**28.17**
Co-training	77.5	*0.421*	*0.831*	0.716	0.181	*28.83*

Figure 5.9: Best model from each approach on labeled Pexels dataset. Best results in **Bold**, Second best in *Italics*.

Figure 5.11: Line charts showing the IoU, Accuracy, and F_β for each of the three semi-supervised approaches. The left column is self-training; the middle column is expectation maximisation; the right column is co-training.

Chapter 6

Conclusions

6.1 Discussion

In this paper, we investigated how using semi-supervised learning techniques - namely self-training, expectation maximisation, and co-training, can help to overcome the issues of high data similarity in the video mirror detection problem. We have constructed two new mirror datasets, implemented three distinct semi-supervised techniques to extend the current SOTA model to allow better adaptability, and tested these new implementations. Experimental results show that each of our semi-supervised techniques outperforms the existing state-of-the-art methods for video mirror detection - having a slight performance gain on the pre-existing VMDD-test dataset, and a significant improvement in the adaptability as measured by the newly created annotated dataset.

6.2 Real-time Inferring and Poster Display

As part of this honours project, we also took part in the University of Edinburgh Informatics Project Day. This involved creating an academic poster which can be found in appendix A as well as creating a real-time demo. To our knowledge, this is the first time that real-time mirror detection has been used with live video - as opposed to using prerecorded and preprocessed videos. This involved a slight modification to the model as in real-time it can no longer take in the third randomly sampled frame, and as opposed to using JPEG images it has to work off of live videos.

Our project won the first place prize for best poster - decided based on the quality of the poster and presentation. This decision was made by many representatives of research institutes from the University of Edinburgh School of Informatics.

6.3 Limitations

There are a lot of additional experiments which we wanted to run but didn't have enough time. These include testing the co-training technique with different image-based models,

Figure 6.1: Occasions where a face is visible in the reflection in a mirror. This is not an exhaustive list of all occurrences.

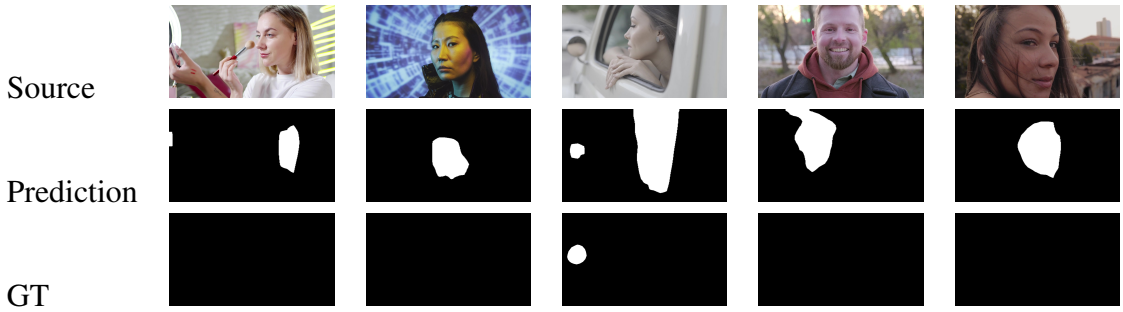


Figure 6.2: Occasions where video mirror detection models predict a face as a mirror region. These predictions are from the original VMD-Net, however, all our semi-supervised models are also prone to this.

testing all models with a different semi-supervised start epoch, and different final epoch values.

The main limitation of our model - which is also present in the existing SOTA model - is that the model learns a correlation between faces and mirrors and so the final model tends to report faces as a mirror region. This is because as one could imagine when collecting a dataset of mirrors, one's reflection would appear in the mirrors in the dataset they're collecting. This can be seen in 6.1 where the face of the researcher collecting the dataset can be seen in the reflection of the mirror. The model then subconsciously learns the correlation between the human face and the mirror region, which in turn ends with the final model reporting non-reflected faces as a mirror region. This can be seen in figure 6.2. The solution to this would simply be to collect a negative dataset (a dataset containing just faces without mirrors and all annotated as non-mirror regions) and add this to the training set. Hence, it wouldn't be too difficult to fix.

6.4 Future Work

An interesting area for future research within mirror detection technology is the exploration of models' performance on concave and convex mirrors. Current methodologies have predominantly focused on flat mirrors, leaving a considerable gap in our understanding of model adaptability to mirrors with various geometries. Investigating how existing image-based and video-based models fare with concave and convex mirrors could provide useful insights. Should these models underperform, it presents an intrigu-

ing opportunity for the design and development of new models specifically tailored for mirrors of diverse shapes, thereby broadening the scope of mirror detection capabilities.

Expanding the detection capabilities to include general reflections represents another significant direction for future work. Presently, mirrors are primarily trained to identify mirrors, neglecting other reflective materials and surfaces such as polished metal, and glass under various lighting conditions, and water. Developing a model that recognises a wider array of reflections would greatly enhance its real-world applicability. By encompassing a broader spectrum of reflective surfaces, such a model would drastically improve the errors caused by reflections in existing CV tasks.

A slightly different direction for future work in the mirror detection domain could be incorporating explainable AI (XAI) techniques into mirror detection models. This would offer a promising pathway to enhance the transparency of these systems. Explaining the reasoning behind a model's classification of specific regions as mirrors would provide valuable insights into the decision-making process. This would help refine and improve algorithms by providing reasons why they fail in particular cases. On top of this, it would also be crucial for real-world applications by allowing users to have more trust in the system by making the operations more accessible to the users. The integration of explainable AI could, therefore significantly improve the accuracy and reliability of mirror detection technologies.

Lastly, the development of efficient video representational learning emerges as a compelling project somewhat related to this one. This research would aim to create methodologies that convert video data into succinct, low-dimensional vectors. Such representations could facilitate the comparison of entire video sequences, making it possible to compute the cosine similarity between any two videos. Given the current challenges associated with the absence of dedicated video representational models, this avenue of research could address a critical need paving the way for a more sophisticated and nuanced analysis of video data.

Each of these future research directions not only aims to expand the existing capabilities within the domain of mirror detection but also seeks to address the broader challenges encountered in computer vision. By prioritising these areas, the field can move towards more comprehensive, accurate, and versatile detection models that are better equipped to understand and interpret the world around us. I suspect each of the future works proposed above could be an appropriate level for future undergraduate or master's level honours projects.

Bibliography

Aws rekognition. https://aws.amazon.com/rekognition/. Accessed: 22-02-2024.

Cc0 1.0 universal (cc0 1.0) public domain dedication. https://creativecommons.org/publicdomain/zero/1.0/legalcode. Accessed: 03-03-2024.

Cifar dataset. https://www.cs.toronto.edu/~kriz/cifar.html. Accessed: 18-10-2023.

Coco dataset. https://cocodataset.org/#home. Accessed: 18-10-2023.

Pexels api documentation. https://www.pexels.com/api/documentation/, a. Accessed: 03-03-2024.

Terms of service - pexels. https://www.pexels.com/terms-of-service/, b. Accessed: 03-03-2024.

V7 — the ai data engine for computer vision generative ai. https://www.v7labs.com/. Accessed: 03-03-2024.

P. Anderson, Q. Wu, D. Teney, J. Bruce, M. Johnson, N. Sunderhauf, I. Reid, S. Gould, and A. van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3674–3683, Los Alamitos, CA, USA, jun 2018. IEEE Computer Society. doi: 10.1109/CVPR.2018.00387. URL https://doi.ieeecomputersociety.org/10.1109/CVPR.2018.00387.

M. Berman, A. Triki, and M. B. Blaschko. The lovasz-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4413–4421, Los Alamitos, CA, USA, jun 2018. IEEE Computer Society. doi: 10.1109/CVPR.2018.00464. URL https://doi.ieeecomputersociety.org/10.1109/CVPR.2018.00464.

Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. *Proceedings of the Annual ACM Conference on Computational Learning Theory*, 10 2000. doi: 10.1145/279943.279962.

Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *ArXiv*, abs/2004.10934, 2020. URL https://api.semanticscholar.org/CorpusID:216080778.

Markus Braun, Sebastian Krebs, Fabian Flohr, and Dariu M. Gavrila. Eurocity persons: A novel benchmark for person detection in traffic scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1844–1861, 2019. doi: 10.1109/TPAMI.2019.2897684.

A. Chang, A. Dai, T. Funkhouser, M. Halber, M. Niebner, M. Savva, S. Song, A. Zeng, and Y. Zhang. Matterport3d: Learning from rgb-d data in indoor environments. In *2017 International Conference on 3D Vision (3DV)*, pages 667–676, Los Alamitos, CA, USA, oct 2017. IEEE Computer Society. doi: 10.1109/3DV.2017.00081. URL https://doi.ieeecomputersociety.org/10.1109/3DV.2017.00081.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014. doi: 10.1109/CVPR.2014.81.

Huankang Guan, Jiaying Lin, and Rynson W.H. Lau. Learning semantic associations for mirror detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5931–5940, 2022. doi: 10.1109/CVPR52688.2022.00585.

Dongsheng Han, Chaoning Zhang, Yu Qiao, Maryam Qamar, Yuna Jung, SeungKyu Lee, Sung-Ho Bae, and Choong Seon Hong. Segment anything model (sam) meets glass: Mirror and transparent objects cannot be easily detected, 04 2023.

Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. pages 2980–2988, 10 2017. doi: 10.1109/ICCV.2017.322.

Ruozhen He, Jiaying Lin, and Rynson W.H. Lau. Efficient mirror detection via multi-level heterogeneous learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(1):790–798, Jun. 2023. doi: 10.1609/aaai.v37i1.25157. URL https://ojs.aaai.org/index.php/AAAI/article/view/25157.

Tianyu Huang, Bowen Dong, Jiaying Lin, Xiaohui Liu, Rynson W.H. Lau, and Wangmeng Zuo. Symmetry-aware transformer-based mirror detection. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. ISBN 978-1-57735-880-0. doi: 10.1609/aaai.v37i1.25173. URL https://doi.org/10.1609/aaai.v37i1.25173.

Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 603–612, 2019. doi: 10.1109/ICCV.2019.00069.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, October 2023.

Fengze Li, Jieming Ma, Zhongbei Tian, Ji Ge, Hai-Ning Liang, Yungang Zhang, and Tianxi Wen. Mirror-yolo: A novel attention focus, instance segmentation and mirror detection model. In *2022 7th International Conference on Frontiers of Signal Processing (ICFSP)*. IEEE, sep 2022. doi: 10.1109/icfsp55781.2022.9925001. URL https://doi.org/10.1109%2Ficfsp55781.2022.9925001.

J. Lin, X. Tan, and R. H. Lau. Learning to detect mirrors from videos via dual correspondences. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9109–9118, Los Alamitos, CA, USA, jun 2023. IEEE Computer Society. doi: 10.1109/CVPR52729.2023.00879. URL https://doi.ieeecomputersociety.org/10.1109/CVPR52729.2023.00879.

Jiaying Lin, Guodong Wang, and Rynson W.H. Lau. Progressive mirror detection. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3694–3702, 2020. doi: 10.1109/CVPR42600.2020.00375.

Haiyang Mei, Bo Dong, Wen Dong, Pieter Peers, Xin Yang, Qiang Zhang, and Xiaopeng Wei. Depth-aware mirror segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3044–3053, June 2021.

Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39(2):103–134, 2000. ISSN 1573-0565. doi: 10.1023/A:1007692713085. URL https://doi.org/10.1023/A:1007692713085.

Avital Oliver, Augustus Odena, Colin Raffel, Ekin D. Cubuk, and Ian J. Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 3239–3250, Red Hook, NY, USA, 2018. Curran Associates Inc.

Edwin Olson. Apriltag: A robust and flexible visual fiducial system. In *2011 IEEE International Conference on Robotics and Automation*, pages 3400–3407, 2011. doi: 10.1109/ICRA.2011.5979561.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020, 2021. URL https://arxiv.org/abs/2103.00020.

Jiaqi Tan, Weijie Lin, Angel X. Chang, and Manolis Savva. Mirror3d: Depth refinement for mirror surfaces. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15985–15994, 2021. doi: 10.1109/CVPR46437.2021.01573.

Xin Tan, Jiaying Lin, Ke Xu, Pan Chen, Lizhuang Ma, and Rynson W.H. Lau. Mirror detection with the visual chirality cue. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3492–3504, 2023. doi: 10.1109/TPAMI.2022.3181030.

Thomas Whelan, Michael Goesele, Steven J. Lovegrove, Julian Straub, Simon Green, Richard Szeliski, Steven Butterfield, Shobhit Verma, and Richard Newcombe. Reconstructing scenes with mirror and glass surfaces. *ACM Trans. Graph.*, 37 (4), jul 2018. ISSN 0730-0301. doi: 10.1145/3197517.3201319. URL https://doi.org/10.1145/3197517.3201319.

Shao-Wen Yang and Chieh-Chih Wang. On solving mirror reflection in lidar sensing. *IEEE/ASME Transactions on Mechatronics*, 16(2):255–265, 2011. doi: 10.1109/TMECH.2010.2040113.

X. Yang, H. Mei, K. Xu, X. Wei, B. Yin, and R. Lau. Where is my mirror? In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8808–8817, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society. doi: 10.1109/ICCV.2019.00890. URL https://doi.ieeecomputersociety.org/10.1109/ICCV.2019.00890.

David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting on Association for Computational Linguistics*, ACL '95, page 189–196, USA, 1995. Association for Computational Linguistics. doi: 10.3115/981658.981684. URL https://doi.org/10.3115/981658.981684.

Oliver Zendel, Katrin Honauer, Markus Murschitz, Martin Humenberger, and Gustavo Fernández Domínguez. Analyzing computer vision data — the good, the bad and the ugly. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6670–6680, 2017. doi: 10.1109/CVPR.2017.706.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5122–5130, 2017. doi: 10.1109/CVPR.2017.544.

Appendix A

Informatics Project Day Poster