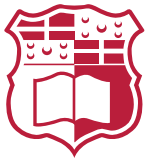
Accurate Name Extraction from News Video Graphics

Andrea Filiberto Lucas

Supervisor: Dr. Dylan Seychell

May 2025

Submitted in partial fulfilment of the requirements for the degree of B.Sc. Information Technology (Hons) in Artificial Intelligence.



L-Università ta' Malta
Faculty of Information &
Communication Technology

Abstract

The growth of video-based media has intensified the need for automated information extraction systems. Graphical elements such as captions and lower thirds frequently contain key identifiers like personal names, yet manual extraction remains inefficient due to design variability across broadcasts. This dissertation tackles the challenge of automatically extracting names from overlaid graphics in news video content, which is complicated by diverse visual styles, typography, and spatial layouts.

This research presents three main contributions. First, the News Graphics Dataset (NGD), a custom dataset of annotated frames sourced from both local and foreign media. It includes traditional broadcasts and social platforms, capturing a wide range of graphical conventions. Second, the Accurate Name Extraction Pipeline (ANEP), a modular framework for name extraction that integrates You Only Look Once (YOLO)v12-based Object Detection (OD), Optical Character Recognition (OCR), and Named Entity Recognition (NER). Third, a comparative evaluation against leading Generative Artificial Intelligence (GenAI) methods, including Google Vision API (GVA) with Gemini 1.5 Pro and Large Language Model Meta AI (LLaMA) 4 Maverick.

Empirical results reveal distinct performance characteristics across approaches. The GVA approach with Gemini 1.5 Pro achieved superior overall performance with an F1 score of 82.22%. However, the ANEP framework demonstrated more balanced precision-recall characteristics (72.92% and 74.44% respectively). It also offered enhanced explainability compared to the generative models.

A complementary survey of 404 respondents validated the research problem. Notably, 59.7% reported difficulties identifying names in news graphics. Additionally, 58.2% confirmed they had paused videos specifically to identify individuals. These findings establish both the technical viability and practical utility of automated name extraction systems. They provide a foundation for future research in multimodal information extraction from broadcast content.

Acknowledgements

I express my sincere gratitude to my parents, Ian and Alexandra Lucas, for their unwavering support throughout my academic journey. Their steadfast encouragement has been fundamental to all my achievements.

I am deeply grateful for my supervisor, Dr. Dylan Seychell, whose expert guidance, constructive feedback and dedicated mentorship proved invaluable. His expertise and patience were instrumental in navigating the research process and completing this dissertation.

Finally, my sincere thanks extend to all participants who contributed to the survey at `aflucas.com/dissertation-form`. Their insights significantly enhanced the depth and validity of this research.

Contents

Abstract i

Acknowledgements ii

Contents vi

List of Figures x

List of Tables xi

List of Abbreviations xiv

1 Introduction 1
- 1.1 Problem Definition . . . 1
- 1.2 Motivation . . . 1
- 1.3 Proposed Solution . . . 2
- 1.4 Aims and Objectives . . . 3
- 1.5 Document Structure . . . 3

2 Background 4
- 2.1 News Video Graphics . . . 4
- 2.2 Identifying Visual Elements in News Videos . . . 5
  - 2.2.1 Frame Sampling and Deduplication . . . 5
- 2.3 Extracting Text from News Video Graphics . . . 6
- 2.4 Validating Extracted Text with NER . . . 7
- 2.5 Emerging Trends in Automated Name Extraction . . . 7

3 Literature Review 8
- 3.1 Datasets for News Video Analysis . . . 8
  - 3.1.1 Existing Datasets and Their Limitations . . . 8
  - 3.1.2 The Need for NGD . . . 9
- 3.2 Computer Vision (CV) Techniques for Video Frame Processing . . . 9
  - 3.2.1 Frame Sampling and Deduplication . . . 9
  - 3.2.2 OD Techniques in Video Frames . . . 9

3.2.3 Preprocessing for Text Extraction . . . 10
3.2.4 Challenges in Broadcast Graphics Analysis . . . 10
3.3 OCR in Video Analysis . . . 10
3.3.1 Overview & Challenges . . . 11
3.3.2 Approaches & Tools . . . 11
3.3.3 Evaluation and Future Directions . . . 12
3.4 Extracting Names Using NER . . . 12
3.4.1 Overview & Relevance . . . 12
3.4.2 Integration of NER in OCR-based Pipelines . . . 13
3.4.3 Models and Architectures for NER . . . 13
3.4.4 Challenges of Noisy OCR Input . . . 13
3.4.5 Multilingual and Cross-lingual NER . . . 14
3.5 GenAI for Name Extraction . . . 14
3.5.1 Large Language Model (LLM)s as Annotation Engines . . . 14
3.5.2 Multimodal Name Extraction with Foundation Models . . . 15
3.5.3 Current Limitations and Hybrid Approaches . . . 15

4 Methodology **16**
4.1 The NGD . . . 16
4.1.1 Data Gathering . . . 16
4.1.2 Frame Extraction . . . 17
4.1.3 Data Annotation . . . 17
4.1.4 Dataset Analysis & Visualisation . . . 17
4.1.5 Data Preprocessing & Augmentation . . . 18
4.2 Object Detection Model Training YOLO . . . 19
4.2.1 Roboflow Training . . . 19
4.2.2 Local Training . . . 19
4.3 The ANEP . . . 20
4.3.1 Pipeline Overview . . . 20
4.3.2 Frame Processing . . . 20
4.3.3 Graphic Detection and Preprocessing . . . 21
4.3.4 Text Extraction . . . 21
4.3.5 Named Entity Extraction & Validation . . . 21
4.3.6 Name Clustering & Timeline Generation . . . 22
4.4 GenAI Application Programming Interface (API)s-Based Video Analysis Pipelines . . . 22
4.4.1 Google Cloud Vision & Gemini 1.5 Pro . . . 22
4.4.2 LLaMA 4 Maverick . . . 23
4.5 ANEP User Interface (UI) . . . 23

4.5.1 Architecture & Component Structure . . . 23
4.5.2 User Experience (UX) Design . . . 24
4.5.3 Dissertation Survey Visualisation . . . 24

5 Evaluation & Results **25**
5.1 Evaluation Metrics . . . 25
5.1.1 Classification Metrics . . . 25
5.1.2 Object Detection Metrics . . . 25
5.2 OD Results . . . 26
5.2.1 Model Comparison . . . 26
5.2.2 OD Evaluation Metrics . . . 26
5.2.3 Model Visualisation . . . 29
5.3 Comparative Name Extraction Performance . . . 32
5.3.1 Evaluation Methodology and Metrics . . . 32
5.3.2 Quantitative Results and Discussion . . . 32
5.3.3 Error Analysis and Limitations . . . 33
5.4 Survey Analysis . . . 35
5.4.1 Demographics and News Consumption Patterns . . . 35
5.4.2 Name Graphics and Viewer Interaction . . . 36
5.4.3 Perceptions of Automated Tools . . . 37

6 Conclusion **38**
6.1 Summary of Contributions . . . 38
6.2 Key Findings and Results . . . 39
6.3 Limitations . . . 39
6.4 Future Work . . . 40

A Dissertation Resources **50**

B Annotation Guidelines **51**

C Data Flow Diagram (DFD) & Flowcharts **53**
C.1 Overview of ANEP . . . 53
C.2 Level 1 DFD of the ANEP UI System . . . 54
C.3 ANEP UI Flowchart . . . 55
C.4 ANEP Flowchart . . . 56
C.5 GVA Flowchart . . . 57
C.6 LLaMA Flowchart . . . 58

D Prompt Templates **59**
D.1 Gemini Prompt for Name Extraction . . . 59

D.2 LLaMA Prompts for OCR 60
D.2.1 Primary Prompt 60
D.2.2 Fallback Prompt 60

E Additional Evaluation Results **61**
E.1 NGD Dataset: Statistical Breakdown 61
E.2 YOLOv12: Further Visualisation Examples 64
E.3 YOLOv12: Training and Evaluation Metrics 64

F NGD Breakdown **69**
F.1 Overview of NGD 69
F.2 Video Downloads by Source 70
F.3 Annotated Frames by Source 71

G Frame Extraction & ANEP UI **72**
G.1 Frame Extraction GUI 72
G.2 ANEP's Web UI 73
G.2.1 Upload Step 73
G.2.2 Model Selection Step 75
G.2.3 Confirm Step 76
G.2.4 Analyse Step 77
G.2.5 Results Step 78
G.2.6 Survey Visualisation 80

List of Figures

Figure 1.1 Example output from ANEP: (**1**) headline and logo graphics are detected using YOLOv12, (**2**) the headline is processed via OCR and NER to extract the name "Robert Prevost," and (**3**) results are saved in JavaScript Object Notation (JSON) format with timestamps 2
Figure 1.2 Web UI icon generated using Generative Pre-trained Transformer (GPT)-4o Image Generation 2

Figure 2.1 Representative samples from the NGD showing annotated news video frames across diverse broadcast sources 4
Figure 2.2 Tesseract OCR pipeline showing thresholding, segmentation, and two-pass text recognition [34] 6

Figure 3.1 Training pipeline for the GLiNER bi-encoder using DeBERTa, BGE, and fine-tuning on NuNER and high-quality examples [81] 14

Figure 4.1 Class distribution within the NGD 18
Figure 4.2 Level 0 DFD showing the high-level interaction between the user, ANEP frontend and back-end modules, and associated data stores 24

Figure 5.1 Evaluation curves for each class: precision–confidence, recall–confidence, precision–recall, and F1–confidence relationships 28
Figure 5.2 Training dynamics of NGD-YOLOv12_v5 showing metric convergence and loss over 110 epochs 29
Figure 5.3 Grad-CAM visualisation for a representative news frame, showing original input (left), activation map (centre), and overlay (right) from Layer 1 of YOLOv12 29
Figure 5.4 Sample 1: Successful detection from TVM news 30
Figure 5.5 Sample 2: Frame extracted from a photo of a Sky News broadcast . . 30
Figure 5.6 Sample 3: Accurate detection from Rai News despite no training exposure 31
Figure 5.7 Sample 4: Generalisation to social media overlay from Daily Mail . . . 31
Figure 5.8 Preferred sources for accessing news content 35
Figure 5.9 Preferred formats for news consumption 36

Figure 5.10 Respondents who reported difficulty reading names in news video graphics 37
Figure 5.11 Trust in AI-based name extraction tools among participants 37

Figure C.1 Simplified workflow of ANEP 53
Figure C.2 Level 1 DFD of ANEP UI 54
Figure C.3 ANEP UI Workflow 55
Figure C.4 ANEP System Architecture 56
Figure C.5 GVA Workflow 57
Figure C.6 LLaMA Workflow 58

Figure E.1 Class co-occurrence matrices by split, showing how frequently each class appears alongside others. 61
Figure E.2 Class distribution across all splits, sorted by total frequency. 62
Figure E.3 Heat maps showing spatial density of object annotations across splits. Strong horizontal bands reflect common placement of on-screen graphics. . . 62
Figure E.4 Histogram showing the distribution of object counts per image in the training split. Most images contain between 3 and 5 graphic elements. 63
Figure E.5 Scatter plots of normalised bounding box centres across dataset splits. Clustered horizontal and vertical alignments reflect standardised broadcast layouts. 63
Figure E.6 Example 1 — BBC broadcast shown on a TV screen. The model accurately detects overlapping graphics despite the indirect angle, though it misclassifies the `Breaking News` banner as a `Lower Third Graphic`. 64
Figure E.7 Example 2 — CNN studio broadcast. Most graphics are correctly detected, but the `Breaking News` label is again misclassified as a `Lower Third Graphic`, likely due to class imbalance in the NGD. 64
Figure E.8 Raw confusion matrix showing class-wise prediction counts for the `NGD-YOLOv12_v5` model. 65
Figure E.9 Normalised confusion matrix, highlighting relative misclassifications and inter-class confusion. 65
Figure E.10 F1 score vs confidence threshold for each class, showing optimal threshold (0.435) achieving 94% global F1. 66
Figure E.11 Precision vs confidence curve across all classes, reflecting class-specific reliability at varying thresholds. 66
Figure E.12 Recall vs confidence curve showing detection sensitivity trends by class. 67
Figure E.13 Precision–recall curves per class, with Mean Average Precision (mAP)@0.5 reaching 95.8% overall. 67
Figure E.14 Scatter plot of precision vs recall across training epochs, colour-coded by epoch number. 68

Figure E.15 F1 score progression over training epochs, showing convergence near 93%. 68

Figure G.1 Frame selection interface showing extracted candidate frames from a TVM news video - Each frame is accompanied by a frame number and the option to replace it, allowing manual refinement prior to analysis 72
Figure G.2 Upload interface in dark mode - Users can drag and drop a video file to initiate the pipeline 73
Figure G.3 Upload interface in light mode - Same functionality presented with an alternative visual theme 74
Figure G.4 A video file has been successfully uploaded to the ANEP platform. Metadata such as filename, size, duration, and format are displayed prior to proceeding to model selection 74
Figure G.5 Model selection interface with four pipelines: ANEP, LLaMA 4 Maverick, Google Cloud Vision & Gemini, and Comparative Analysis, each with a brief description 75
Figure G.6 Model comparison table showing the relative performance of different name extraction pipelines. Metrics include speed, accuracy, suitability, and API requirements 75
Figure G.7 Confirmation screen summarising selected analysis details, including video filename, chosen model, estimated processing time, and expected outputs 76
Figure G.8 The video analysis step in progress, showing the current completion percentage and processing time - Users may cancel the analysis at any stage 77
Figure G.9 The interface following an analysis cancellation - The UI displays an error state with logs and troubleshooting tips to help resolve common issues 77
Figure G.10 Tabular view of detected names. Includes name, timestamp, and confidence score 78
Figure G.11 Compact result view displaying the name detected, initial timestamp, and confidence - The UI offers a full-screen toggle for easier inspection . . . 78
Figure G.12 Structured JSON view of the name detection results, including timing information, model metadata, and entity confidence scores 79
Figure G.13 Metadata tab showing model information, detection overview, and file metadata - Useful for exporting or archiving detection reports 79
Figure G.14 Visualisation controls interface allowing users to toggle chart types, sort order, and display settings for survey data presentation 80
Figure G.15 Bar chart showing news consumption frequency for males aged 18-24, enabling demographic subgroup analysis 81

Figure G.16 Line chart of responses with export options for chart copying, filtered CSV data, and full dataset download 81
Figure G.17 Pie chart representation showing proportional responses to the question on news consumption frequency 82
Figure G.18 Tabular survey result view with detailed response breakdown, count, and percentage 82

List of Tables

Table 4.1 Data augmentation settings used during training, calibrated to preserve visual realism while increasing variability 19
Table 4.2 Optimised ANEP parameters for efficient frame sampling, redundancy reduction, and high-confidence detection across YOLO, OCR, and NER stages 21

Table 5.1 Comparison of YOLO-based OD models trained on the NGD. The best-performing model, **YOLOv12*(m)*** was locally trained and led across all metrics. 27
Table 5.2 Performance of name extraction models in terms of precision, recall, F1 score, and processing time 32

Table F.1 Summary of video counts grouped by domain 69
Table F.2 Summary of annotated image frames per domain in the NGD 69
Table F.3 Number of videos downloaded per source, grouped by content domain 70
Table F.4 Number of annotated frames per source, grouped by content domain . 71

List of Abbreviations

AI Artificial Intelligence.

ANEP Accurate Name Extraction Pipeline.

API Application Programming Interface.

ASR Automatic Speech Recognition.

BERT Bidirectional Encoder Representations from Transformers.

BoW Bag of Words.

CER Character Error Rate.

CLAHE Contrast Limited Adaptive Histogram Equalisation.

CNN Convolutional Neural Network.

COCO Common Objects in Context.

CoT Chain of Thought.

CPU Central Processing Unit.

CRF Conditional Random Field.

CUDA Compute Unified Device Architecture.

CV Computer Vision.

DCT Discrete Cosine Transform.

DFD Data Flow Diagram.

DL Deep Learning.

FP False Positive.

FPN Feature Pyramid Network.

FPS Frame Per Second.

GenAI Generative Artificial Intelligence.

GLiNER Generalist Language-independent Named Entity Recognition.

GPT Generative Pre-trained Transformer.

GPU Graphics Processing Unit.

GUI Graphical User Interface.

GVA Google Vision API.

IoU Intersection over Union.

IP Internet Protocol.

iRoPE Interleaved Rotary Position Embedding.

JPEG Joint Photographic Experts Group image format.

JSON JavaScript Object Notation.

KB Knowledge Base.

LLaMA Large Language Model Meta AI.

LLM Large Language Model.

LSTM Long Short-Term Memory.

mAP Mean Average Precision.

MoE Mixture of Experts.

MPS Metal Performance Shaders.

MultiCoNER Multilingual Complex Named Entity Recognition.

NER Named Entity Recognition.

NGD News Graphics Dataset.

NLP Natural Language Processing.

NMS Non-Maximum Suppression.

OCR Optical Character Recognition.

OD Object Detection.

ORB Oriented FAST and Rotated BRIEF.

PSNR Peak Signal-to-Noise Ratio.

R-CNN Region-based Convolutional Neural Network.

RoBERTa Robustly Optimised BERT Pretraining Approach.

ROI Region of Interest.

RPN Region Proposal Network.

SIFT Scale-Invariant Feature Transform.

TrOCR Transformer-based Optical Character Recognition.

TV Television.

TVIA Television Internet Archive.

UI User Interface.

UX User Experience.

WER Word Error Rate.

YOLO You Only Look Once.

---

1 Introduction

This chapter outlines the problem definition, motivation, proposed solution, and primary aims and objectives of this dissertation.

1.1 Problem Definition

The contemporary information landscape presents an unprecedented challenge of data abundance that frequently overwhelms users [1]. This information overload is particularly acute in the media industry, where comprehensive awareness is essential for professional effectiveness [2]. The inherent diversity, volume and complexity of modern content significantly exacerbate this challenge [1, 2]. As short news videos have emerged as a dominant and increasingly popular media format [3], there exists a critical need for sophisticated tools capable of efficiently extracting and highlighting key information. Without such analytical capabilities, significant insights remain obscured, substantially limiting audiences' capacity to thoroughly analyse, interpret and leverage content effectively [4–7].

1.2 Motivation

Contemporary news media presents a complex visual environment wherein critical textual information is frequently embedded within dynamic graphical overlays. Lower-thirds, captions, and headlines serve as primary conduits for conveying essential contextual cues, including the identification of individuals featured in the video content. Despite the ubiquity of such elements, their inherent variability presents significant challenges for automated information extraction.

Existing research in media analysis has made substantial advancements in visual information processing [8–12]. However, the automated extraction of names from video graphics remains a largely unexplored domain. The various design approaches employed by different news organisations, ranging from stylistic variations in typography to inconsistent placement and transient display durations, create substantial obstacles for developing generalised extraction methodologies [13, 14].

This dissertation advances automatic detection and recognition of names from broadcast video graphics to enrich media analysis pipelines. It extends systems like Seychell et al.'s [15] for television content summarisation by introducing a generalisable mechanism for visual named entity identification. These contributions align with recent multimodal information extraction developments [16], integrating CV, OCR, and NER for scalable media intelligence systems.

1.3 Proposed Solution

This research introduces an automated system for extracting names from graphical elements in news videos. The proposed framework, designated as the Accurate Name Extraction Pipeline (ANEP), integrates three complementary technologies: YOLO, an advanced Convolutional Neural Network (CNN), for precise detection of graphical components within video frames; OCR technology, optimised to extract textual information across diverse font styles and varying resolutions; and NER, implemented to identify and validate personal names with high accuracy. To assess efficacy, the ANEP will undergo rigorous comparative evaluation against two state-of-the-art generative GenAI models using identical video inputs. This evaluation framework employs multiple performance metrics, namely precision, recall, F1-score and processing time, providing a comprehensive analysis of the relative strengths and limitations inherent in both traditional and generative methodological approaches.

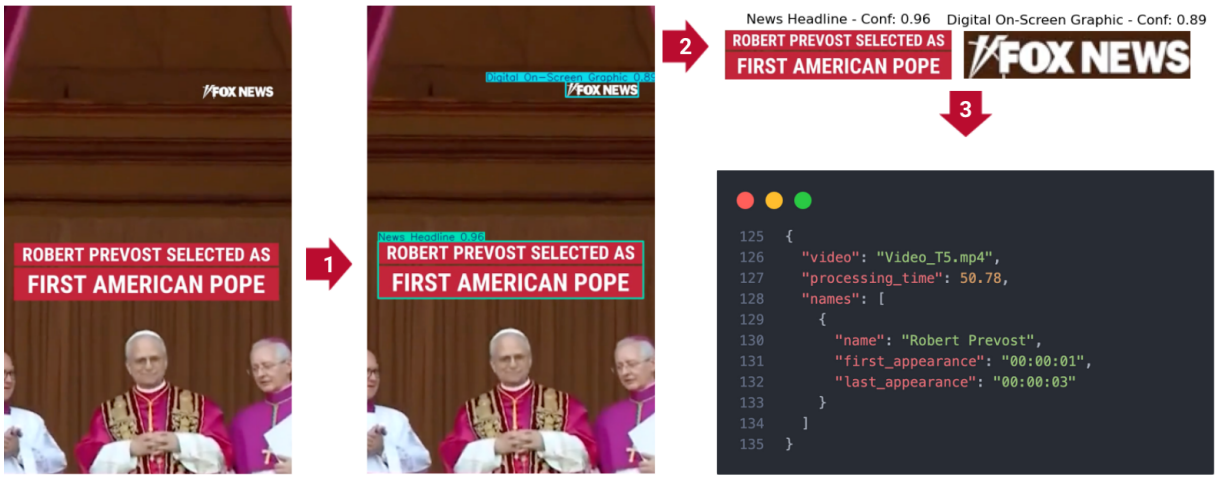


Figure 1.1 Example output from ANEP: (**1**) headline and logo graphics are detected using YOLOv12, (**2**) the headline is processed via OCR and NER to extract the name "Robert Prevost," and (**3**) results are saved in JSON format with timestamps

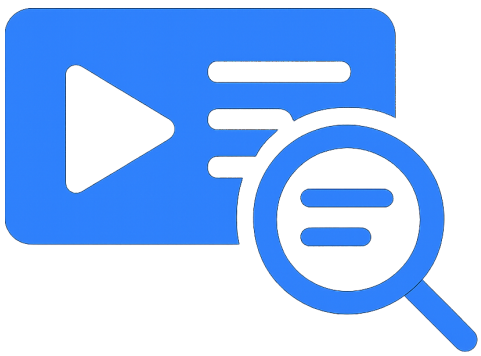


Figure 1.2 Web UI icon generated using GPT-4o Image Generation

1.4 Aims and Objectives

The primary aim of this research is to develop a system for the precise extraction of names from news video graphics and to rigorously evaluate its efficacy against cutting-edge generative GenAI models. Ultimately, the research will provide a structured analysis of the extracted names, enhancing media analysis and accessibility. This will be achieved through the following objectives:

1. **Build the NGD** incorporating diverse graphical elements from multiple news sources for robust YOLO model training.
2. **Design and implement a comprehensive pipeline (ANEP)** that seamlessly integrates CNN architecture, OCR capabilities, and NER methodologies to optimise name extraction from graphical video content with exceptional accuracy.
3. **Implement Comparative Analysis Framework** integrating optimised GenAI API calls and generating comprehensive performance metrics across all extraction methodologies.

1.5 Document Structure

This dissertation progresses from theory to implementation and evaluation, with chapters structured to coherently address the automated name extraction challenge.
Chapter 2 – Background outlines the technological context, covering news graphics, YOLO-based OD, OCR, NER, and emerging GenAI methods that inform the comparative approach used in this study.
Chapter 3 – Literature Review critically examines prior work on datasets, CV, OCR, and NER, concluding with GenAI approaches and highlighting research gaps addressed by this dissertation.
Chapter 4 – Methodology details the technical framework, including the NGD, YOLO and ANEP development, two GenAI-based pipelines, and a unified web interface supporting all extraction methods.
Chapter 5 – Evaluation & Results presents empirical analysis of OD performance, compares traditional and generative methods, and integrates survey findings that highlight the practical relevance of automated name extraction in modern media contexts.
Chapter 6 – Conclusion summarises the main contributions, including the NGD dataset, ANEP performance, and GenAI comparisons, while acknowledging limitations and outlining future work in multimodal and real-time video analysis.

2 Background

This chapter outlines the technologies underpinning the dissertation's pipeline, focusing on challenges posed by variability in news video graphics. It highlights the application of YOLO for graphical element detection and Region of Interest (ROI)s identification, OCR for text extraction, and NER for entity validation and contextualisation. Additionally, it explores emerging trends in GenAI as a comparative approach for evaluating and enhancing traditional extraction methods.

2.1 News Video Graphics

News video graphics serve as essential information carriers in modern broadcasting, providing viewers with critical context through visual elements such as lower thirds, headlines, digital on-screen graphics (DOGs), and information boxes that display names, locations, titles and event details [13, 14, 17–19]. These graphical components have become ubiquitous across traditional television broadcasts, digital streaming platforms and social media channels [18, 19], functioning both as narrative anchors and audience engagement tools through strategic visual reinforcement [18]. Their importance is particularly evident during live reporting and breaking news, where rapid communication of accurate information is vital. Moreover, universally recognised symbols and imagery enable these graphics to overcome language barriers, facilitating communication with global audiences [13, 14].

However, the benefits of news video graphics also introduce challenges for automated analysis. The absence of industry-wide design standards [20, 21] results in considerable variation across broadcasters, with each outlet adopting distinctive styles, layouts and formatting conventions, complicating consistent detection and extraction [20]. This variability is further complicated by the dynamic nature of broadcast graphics, including animated transitions, real-time updates and moving text, all of which increase the complexity of reliable identification and automated analysis.

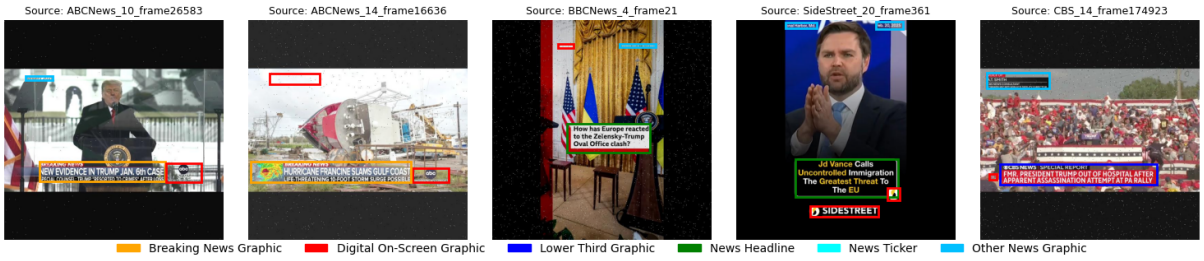


Figure 2.1 Representative samples from the NGD showing annotated news video frames across diverse broadcast sources

2.2 Identifying Visual Elements in News Videos

Detection frameworks based on CNN architectures have shown strong performance in this domain. Among them, YOLO stands out for combining localisation and classification in a single, unified process [22]. By dividing images into grids and predicting bounding boxes and class probabilities simultaneously, YOLO balances real-time performance with accuracy, making it well-suited to the fast-paced nature of news broadcasting [23–25]. Over successive versions, culminating in YOLOv12, the architecture has incorporated advanced features such as Transformer-based modules that improve long-range feature modelling. These architectural variants offer different performance and efficiency trade-offs, enabling more accurate detection of smaller or overlapping elements within frames [26, 27], addressing the dense and cluttered visual environments often found in news broadcasts.

In this context, detected graphical elements are treated as ROIs, which are subsequently cropped and analysed to extract embedded textual information. Accurate identification of these ROIs is crucial, as errors at this stage propagate through the remainder of the extraction pipeline.

Robust detection systems rely on high-quality, annotated datasets that capture the visual diversity of news graphics. These must reflect variations in typography, colour, layout, branding, and challenges like motion blur, occlusion, and lighting changes. To improve generalisation and resilience, training often includes data augmentation such as cropping, brightness/exposure adjustments, and synthetic noise [28].

2.2.1 Frame Sampling and Deduplication

Efficient processing of video content necessitates intelligent frame sampling and deduplication techniques to reduce computational overhead while preserving critical information. Perceptual hashing algorithms, including those based on the Discrete Cosine Transform (DCT) and feature detection methods such as Oriented FAST and Rotated BRIEF (ORB), facilitate identification of visually distinct frames whilst avoiding redundant processing of similar content [29]. These techniques are especially valuable when analysing news broadcasts, which often contain static scenes interspersed with rapidly changing content.

Effective deduplication strategies typically incorporate similarity thresholds that balance processing efficiency against information loss. Additionally, multi-level approaches combining hash-based filtering with more computationally intensive feature comparison can achieve robust frame selection in varying visual conditions. Such techniques prove essential for practical implementations of video analysis systems, where processing constraints necessitate selective frame evaluation [30].

2.3 Extracting Text from News Video Graphics

OCR refers to the process of converting text embedded in images into a machine-readable format [31, 32]. Modern OCR systems typically begin with pre-processing steps such as greyscale conversion, contrast enhancement using techniques like Contrast Limited Adaptive Histogram Equalisation (CLAHE), adaptive thresholding, morphological filtering and noise reduction, all designed to improve the clarity of text regions [32]. Following this, text is segmented into individual characters or words for feature extraction and classification [31].

The performance of OCR can be negatively affected by real-world challenges, including varying fonts, low contrast, motion blur and the presence of similar-looking characters [33]. To mitigate these problems, advanced systems apply multiple pre-processing strategies alongside post-processing techniques, including dictionary-based corrections, confidence thresholding and contextual text reconstruction, helping to refine extracted outputs and improve their semantic accuracy [31, 33].

Open-source OCR solutions such as Tesseract provide accessible frameworks for text extraction, whilst cloud-based services like GVA offer enhanced performance through extensively trained models, albeit with potential limitations in offline environments. Parallel application of multiple OCR approaches with confidence-based selection can further improve extraction reliability across diverse graphical styles.

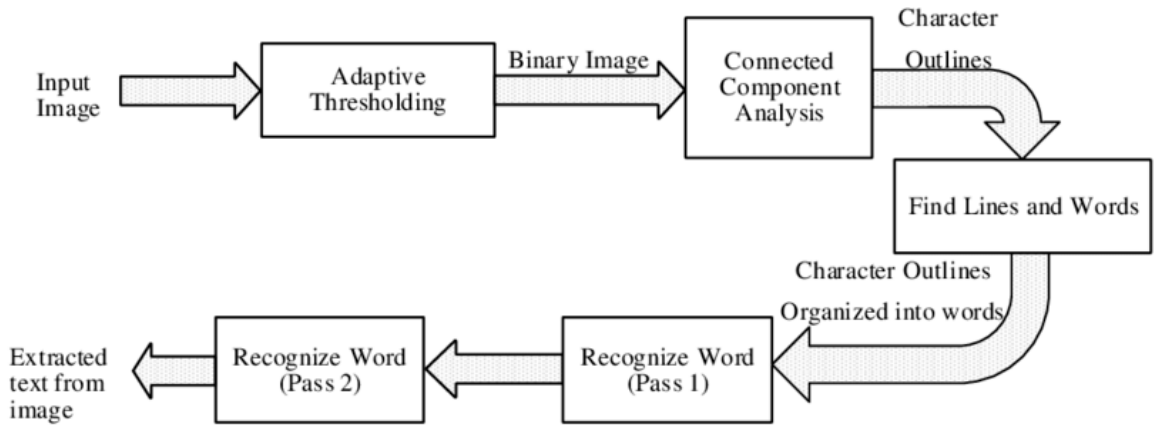


Figure 2.2 Tesseract OCR pipeline showing thresholding, segmentation, and two-pass text recognition [34]

2.4 Validating Extracted Text with NER

NER, a Natural Language Processing (NLP) subfield, identifies and classifies semantic entities within text [35, 36]. Modern systems employ sophisticated Deep Learning (DL) techniques, to handle real-world language variability and complexity [36, 37]. These models analyse syntax and meaning, detecting named entities even in noisy or unstructured text [36].
Lightweight frameworks such as spaCy offer fast and efficient tokenisation and entity recognition pipelines [36]. Extensions like GliNER introduce zero-shot learning approaches into this ecosystem, enabling flexible entity recognition across new domains without extensive retraining. Meanwhile, transformer-based NER models, trained and fine-tuned on large datasets such as CoNLL-2003, improve performance by using contextual embeddings to resolve ambiguities.
Ensemble methods combine multiple NER models to improve reliability and precision. Post-processing through dictionary validation, confidence scoring, and contextual checks refines results [38, 39]. Advanced clustering with fuzzy matching and embedding-based similarity consolidates entity variants, enhancing downstream utility.

2.5 Emerging Trends in Automated Name Extraction

Recent developments in GenAI have further advanced text and name extraction from visual content. Cloud-based services, such as the GVA, leverage large-scale DL architectures to deliver highly robust OCR capabilities across a wide range of graphic styles, offering alternatives to conventional systems.
Additionally, contemporary LLMs like Gemini 1.5 Pro and LLaMA 4 Maverick, trained on vast text corpora, increasingly validate and contextualise extracted text. Their ability to reason over incomplete, noisy, or ambiguous inputs and identify entities through contextual cues surpasses traditional NER approaches. However, challenges including training data bias and reduced model transparency require careful consideration for sensitive tasks [40, 41].
Progressive API accessibility for these advanced models facilitates their integration into practical analysis pipelines, albeit with considerations regarding rate limiting, computational costs and service availability. Hybrid systems combining traditional approaches with GenAI capabilities represent a promising direction for robust name extraction in broadcast media analysis.

3 Literature Review

This chapter reviews literature across five areas relevant to name extraction from news video graphics. Section 3.1 examines video datasets and their limitations in annotating graphical overlays. Section 3.2 covers CV techniques for sampling, deduplication, object detection, and preprocessing. Section 3.3 addresses OCR for extracting text from dynamic video. Section 3.4 explores NER models and their use in noisy OCR pipelines. Section 3.5 discusses GenAI, including LLMs and hybrid methods, for enhancing name extraction from complex visual content.

3.1 Datasets for News Video Analysis

Whilst numerous datasets exist for general-purpose video analysis, few address the specific challenges of name extraction from overlaid graphics in news broadcasts. Many corpora either lack fine-grained annotations of text elements or are not publicly available, limiting their practical utility in OCR-NER pipelines.

3.1.1 Existing Datasets and Their Limitations

The *TV News Archive*[1], maintained by the Internet Archive, offers over 290,000 hours of annotated foreign cable news broadcasts, including video, audio, and closed captions [42]. Whilst it provides valuable temporal metadata, the dataset lacks structured labelling for on-screen graphical elements such as tickers, lower thirds, or banners, which are critical components for reliable name extraction from visual media. Lee et al. [43] developed a small-scale dataset of Korean TV interview segments containing overlaid name lines. Although useful for evaluating overlay detection techniques, the dataset remains unpublished and was constructed solely for internal testing, restricting reproducibility.

Moreover, the TNB Database by Kannao and Guha [44] contains 360 hours of annotated Indian news broadcasts but lacks bounding box annotations for visual text and structured graphic region classification. Similarly, legacy benchmarks like TRECVid provide curated news footage with partial metadata [43] but lack frame-level annotations distinguishing graphical overlays from scene text—a critical requirement for NER pipelines in broadcast contexts.

[1] `https://archive.org/details/tv`

3.1.2 The Need for NGD

Given these limitations, this dissertation introduces the NGD, a purpose-built corpus of annotated frames from both traditional news broadcasts, sourced via the Television Internet Archive (TVIA) and comprising local Maltese stations (TVM, One News, Net News), and TikTok-based social media news videos covering both local and foreign content. Survey results, as disused in section 5.4, confirm social media platforms as dominant sources of video news consumption, necessitating a dataset reflecting both legacy and emerging formats.

3.2 CV Techniques for Video Frame Processing

This section examines CV techniques applied to individual video frames in news broadcast analysis, covering strategies for efficient frame selection, OD architectures, image preprocessing for text extraction, and practical challenges in analysing dynamic on-screen graphics.

3.2.1 Frame Sampling and Deduplication

Effective video frame processing requires robust frame sampling and deduplication techniques to manage computational resources efficiently whilst maintaining high-quality visual information. Hammoudeh et al. [45] emphasise the importance of selecting representative frames to reduce redundancy. A common approach involves extracting frames at regular intervals, such as one Frame Per Second (FPS), balancing computational efficiency with information retention [45, 46].
Deduplication techniques, such as the Bag of Words (BoW) method adapted from textual and image retrieval applications, identify near-duplicate frames effectively. The BoW approach employs robust keypoints identified via Scale-Invariant Feature Transform (SIFT), which offers resilience to rotation, scaling, and translation [47, 48]. As demonstrated by Gupta et al. [48], these keypoints are clustered into visual vocabularies using k-means++, with frames represented as histograms of visual words, enabling efficient comparisons through Euclidean distance and Peak Signal-to-Noise Ratio (PSNR) [47].

3.2.2 OD Techniques in Video Frames

OD techniques significantly influence video frame analysis effectiveness. YOLO and Faster Region-based Convolutional Neural Network (R-CNN) represent two prevalent methods, each with distinct strengths. YOLO, a single-stage detector, directly predicts bounding boxes and class probabilities in one forward pass, achieving rapid inference

speeds suitable for real-time applications [49–54]. In their performance evaluation of recent YOLO variants, Tian et al. [54] report that YOLOv12 integrates attention mechanisms, multi-scale prediction, and adaptive loss functions, substantially enhancing detection accuracy without compromising processing speed.
In contrast, Faster R-CNN employs a two-stage approach comprising a Region Proposal Network (RPN) to generate candidate regions, followed by classification and bounding box refinement stages. According to Ren et al. [55], whilst this architecture operates more slowly, it provides superior detection accuracy, particularly in complex or occluded scenarios [46, 49, 51]. Feature Pyramid Network (FPN) can further enhance its performance by addressing multi-scale detection challenges [49].

3.2.3 Preprocessing for Text Extraction

Image preprocessing is essential for reliable OCR in video analysis pipelines. In CV-driven video frame processing, preprocessing enhances frame quality before text detection or recognition occurs. Techniques such as CLAHE, adaptive thresholding, and de-noising improve contrast and suppress artefacts [56, 57]. Kamwal [57] discusses additional preprocessing steps including geometric correction, de-skewing, and morphological operations, which stabilise and normalise incoming video frames. These adjustments prove particularly valuable before passing frames to object detection models or text region localisation components.

3.2.4 Challenges in Broadcast Graphics Analysis

Despite these advances, applying CV techniques to dynamic or animated video graphics remains challenging. Hammoudeh et al. [45] observe that issues such as occlusion, animation transitions, and broadcaster-specific graphic styles continue to hinder consistent detection. Whilst data augmentation strategies and multi-scale predictions show promise, they require further refinement to achieve consistent accuracy across diverse broadcast scenarios [45, 50, 58].

3.3 OCR in Video Analysis

This section examines OCR techniques applied to video, with particular emphasis on extracting overlay text from news broadcasts and addressing the visual and temporal challenges inherent in dynamic media.

3.3.1 Overview & Challenges

OCR is fundamental to extracting embedded textual content from video frames, particularly in news broadcasts where overlay text conveys essential information. As noted by Ashraf et al. [59] and Smith [60], Tesseract remains one of the most influential open-source OCR engines, originally developed at HP Labs and later enhanced by Google. Tesseract employs connected component analysis, line segmentation, and adaptive recognition to detect text of varying fonts, orientations, and scales [60].

Unlike static document processing, applying OCR to video frames presents unique challenges rooted in the dynamic nature of video content. Text within video suffers from motion blur, low resolution, geometric distortions, and frequent occlusion due to overlapping graphics [59, 61, 62]. In stylised news broadcasts, overlay text may appear with animations, unconventional fonts, or transparent backgrounds, complicating detection and recognition [44, 63]. Natural scene videos compound these issues through cluttered backgrounds, varied lighting, and camera motion [64].

Additionally, video text varies in orientation, flickers across frames, or appears intermittently, requiring OCR systems to operate reliably across temporal and spatial inconsistencies. These factors render traditional document-based OCR models insufficiently robust without adaptation. To address these obstacles, various image preprocessing and enhancement techniques are employed before recognition, as discussed in Section 3.2.3.

3.3.2 Approaches & Tools

Smith [60] details Tesseract's segmentation pipeline, which employs line detection and moment-based skew correction. Despite effectiveness on scanned documents, these mechanisms falter when confronted with frame-to-frame inconsistencies typical of broadcast media. To address sequential variability, recent work introduces DL-based models such as multi-dimensional Long Short-Term Memory (LSTM) and transformer-based architectures like Transformer-based Optical Character Recognition (TrOCR) [65, 66]. These models integrate image encoding and text decoding in an end-to-end framework, outperforming traditional pipelines on distorted or unconstrained inputs.

Comparative evaluations demonstrate varying strengths amongst available tools. EasyOCR, whilst slower than Tesseract, handles small or degraded scene text more accurately [67]. Cloud-based services such as GVA's OCR offer higher precision on diverse fonts and layouts, though they raise concerns regarding scalability, data privacy, and platform dependency [68]. Domain-specific applications have shown particular promise; Li et al. [69] propose an integrated pipeline for extracting titles from news video using CNN-based detection and classification, demonstrating the potential of specialised DL models in structured overlay analysis. Bhojne [70] highlights the value of overlay text for

structuring searchable video databases, whilst Muthusundari et al. [66] advocate combining Artificial Intelligence (AI)-driven recognition with preprocessing enhancements for constrained visual conditions.
Ensemble OCR strategies remain largely under-explored. Most systems apply a single engine post-detection, although Tesseract's dual-pass recognition and frame aggregation act as internal ensemble-like mechanisms [63].

3.3.3 Evaluation and Future Directions

To measure OCR efficacy in video settings, Nguyen et al. employ Character Error Rate (CER), Word Error Rate (WER), and video-specific metrics such as sequence precision-recall [61]. Modified Tesseract models trained on domain-specific text from Television (TV) news achieved substantial improvements, reducing character error from 20.97% to 4.99% and word error from 56.94% to 7.04% [63]. However, persistent challenges including false positives from background artefacts, missed detections due to animation effects, and limited labelled datasets for broadcast overlay text hinder wider generalisation [61, 62].
Modern systems integrate OCR with NLP tasks like NER for entity extraction from overlay text. Research [43, 69] demonstrates these pipelines enhance applications including person indexing, speaker attribution, and event segmentation. As transformer-based OCR matures alongside larger annotated corpora, visual-semantic analysis interaction promises improved reliability and precision in OCR-driven video understanding.

3.4 Extracting Names Using NER

This section examines NER in news video analysis, detailing its integration with OCR pipelines, model evolution, challenges posed by noisy inputs, and approaches for multilingual content.

3.4.1 Overview & Relevance

NER is a core NLP task identifying and categorising text spans denoting real-world entities such as persons, organisations, locations, and dates [71–73]. In news video analysis, NER proves instrumental in extracting overlaid person names from OCR output, enabling downstream tasks like speaker attribution and semantic indexing [73, 74]. As Damnati et al. [74] note, broadcast systems often rely on NER to annotate individuals introduced on-screen, particularly when prior identity models are unavailable. However, operating on transcriptions from OCR or Automatic Speech Recognition (ASR) introduces significant noise [75, 76].

3.4.2 Integration of NER in OCR-based Pipelines

In typical video pipelines, OCR detects overlaid text and converts it to machine-readable form, after which NER identifies entity spans [73, 74, 77]. Systems may enhance this process by incorporating features such as bounding box coordinates, font styles, and confidence scores from the OCR engine [74]. These auxiliary signals prove particularly valuable in noisy and low-resolution CV frames, where textual content is often incomplete or distorted. Entity linking to external Knowledge Bases (KBs) may follow, improving disambiguation and enabling metadata enrichment [74].

3.4.3 Models and Architectures for NER

Various NER models exist, spanning traditional and DL paradigms. Conditional Random Fields (CRFs) remain effective for sequence labelling due to their contextual dependency modelling [78], frequently deployed in dual-CRF configurations with one for entity tagging and another for associating names with visual clusters [74]. DL approaches, including LSTMs and CNNs, demonstrate strong performance on OCR-derived text, capturing both syntactic and visual context [79, 80].
Transformer-based models such as Bidirectional Encoder Representations from Transformers (BERT), Robustly Optimised BERT Pretraining Approach (RoBERTa), and Generalist Language-independent Named Entity Recognition (GLiNER) achieve state-of-the-art results, particularly on large annotated corpora [37, 72, 75]. Devlin et al. [37] highlight BERT's effectiveness for context-sensitive NER, whilst Zaratiana et al. [75] demonstrate GLiNER's robustness in zero-shot multilingual setups.
Moreover *spaCy*, a widely-used industrial NLP toolkit, provides lightweight yet effective NER performance, particularly when fine-tuned on domain-specific corpora [73]. In contrast, autoregressive models like GPT underperform on entity disambiguation and composite spans [79, 80].

3.4.4 Challenges of Noisy OCR Input

OCR-derived text often suffers degradation due to layout distortions, artefacts, and recognition errors [71, 74, 76]. Even minor increases in WER can reduce NER performance by up to 0.5 points [74]. Such degradation is exacerbated in broadcast settings by cluttered visuals, inconsistent text overlays, and animation effects [78]. Marrero et al. [71] note that ambiguity regarding what constitutes a named entity further complicates reliable extraction.
To address these limitations, recommended practices include using multiple OCR hypotheses, integrating spatial and confidence metadata into the NER model, applying

post-OCR corrections, and leveraging domain-specific name lists during decoding [71, 74, 76]. Systems may also discard entities not reliably linked to known identities to prevent propagation of recognition errors [74].

3.4.5 Multilingual and Cross-lingual NER

Multilingual news broadcasts require robust cross-lingual NER capabilities. Pre-trained models like XLM-RoBERTa and mDeBERTa demonstrate strong multilingual generalisation [72, 76], whilst GLiNER, trained on a multilingual variant of *spaCy*, achieves competitive results on the Multilingual Complex Named Entity Recognition (MultiCoNER) benchmark across 11 languages in zero-shot mode [75]. However, OCR noise complicates multilingual inference, particularly in scripts with limited labelled data or differing orthographic conventions [72, 80]. Pakhale [72] emphasises the need for language-aware preprocessing and domain-specific tuning to handle linguistic diversity effectively.

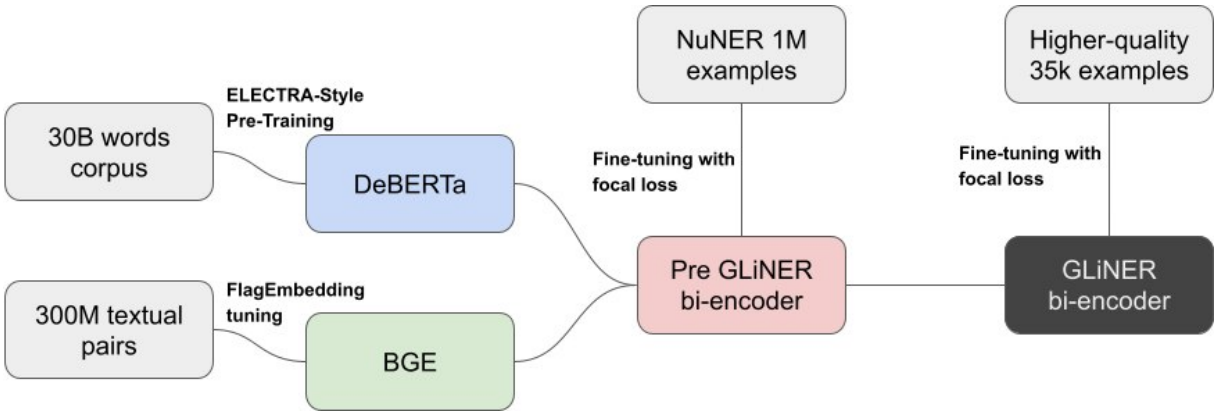


Figure 3.1 Training pipeline for the GLiNER bi-encoder using DeBERTa, BGE, and fine-tuning on NuNER and high-quality examples [81]

3.5 GenAI for Name Extraction

This section examines GenAI for name extraction across three key dimensions: LLMs as annotation engines in NER pipelines; recent multimodal foundation models for direct name extraction from visual and audio sources; and limitations necessitating hybrid approaches combining generative and classical techniques.

3.5.1 LLMs as Annotation Engines

Recent GenAI advancements have substantially reshaped NER, particularly in resource-constrained or noisy contexts such as news video analysis. Rather than replacing clas-

sical NER architectures, LLMss like GPT-4o increasingly serve for data annotation and distillation [82]. In this paradigm, LLMs label entities in raw textual data using structured few-shot or Chain of Thought (CoT) reasoning [82]. As Huang et al. [82] detail, CoT prompting effectively mitigates hallucinations and output instability whilst improving entity classification in noisy categories. The resulting annotated corpora fine-tune smaller, task-specific models such as BERT and RoBERTa, enabling efficient deployment in restricted environments without compromising performance [82].

3.5.2 Multimodal Name Extraction with Foundation Models

Concurrently, recent multimodal LLM architectures such as Gemini 1.5 Pro [83–85] and LLaMA 4 [86–88] demonstrate capabilities for direct name extraction from visual and auditory input. According to Google's Gemini team [84], these models support cross-modal reasoning across text, image, audio, and video, achieving high performance on OCR-related benchmarks without external engines [85]. Gemini 1.5 Pro processes and retrieves content across contexts up to 10 million tokens, allowing holistic analysis of extended transcripts and video streams [84].
Meanwhile, the LLaMA 4 series introduces sparse Mixture of Experts (MoE) computation and the Interleaved Rotary Position Embedding (iRoPE) attention mechanism, as described by Meta AI [87] and analysed by Shukla [88], improving scalability and enabling native multimodal integration. These innovations build upon the Transformer foundation originally introduced by Vaswani et al. [89], marking a substantial advance in generative inference.

3.5.3 Current Limitations and Hybrid Approaches

As Huang et al. [82] highlight, models like Gemini and LLaMA, though powerful, remain computationally intensive and prone to generative instability. Their application often requires meticulous prompt engineering, model tuning, and post-hoc validation to ensure consistent and accurate entity extraction. Furthermore, deployment in real-time video pipelines faces constraints from latency, resource overheads, and vulnerability to input noise [82]. Hybrid pipelines, where GenAI performs annotation or refinement while conventional NER models handle final inference, currently offer the most effective strategy for name extraction in broadcast video domains.

---

4 Methodology

This chapter outlines the methodology, starting with the creation, annotation, and preprocessing of the NGD, followed by the implementation of OD models using various YOLO architectures. It then covers the development of the ANEP, two alternative GenAI-based systems, and the shared UI. Each component is examined with emphasis on technical implementation, architectural decisions, and performance optimisation.

4.1 The NGD

This section details the construction, annotation, preprocessing, and analysis of the NGD, a specialised dataset developed for detecting and classifying news graphics in broadcast video frames (see Appendix F for a further NGD breakdown). The complete annotated dataset is publicly accessible at: `https://universe.roboflow.com/ict3909-fyp/news-graphic-dataset`.

4.1.1 Data Gathering

The NGD was constructed through systematic collection of news videos from diverse sources. Two structured JSON manifest files were created: `TKTK_urls.json` for TikTok-sourced content and `TVIA_urls.json` for traditional broadcasts. Local content was acquired from the official on-demand platforms of national broadcasters, whilst international material came from the TVIA. All videos were downloaded at the maximum available resolution (720p–1080p) to ensure optimal quality for detection tasks.
Distribution analysis using the `Analysis.ipynb` notebook verified that no single broadcaster constituted more than 12.5% of the total corpus, ensuring representational balance across differing graphical styles.
Two custom Python scripts managed the download process: `TKTK_downloader.py` for short-form content and `TVIA_downloader.py` for longer broadcasts. For videos exceeding standard download parameters, the latter implemented an adaptive segmentation strategy dividing content into 185-second fragments—a duration empirically determined to optimise bandwidth stability whilst minimising segment count. These fragments were subsequently reconstructed with validation hashing to ensure integrity.

4.1.2 Frame Extraction

Another Python tool was developed to extract frames containing news graphics, implementing a hybrid approach combining algorithmic selection with human verification. The extraction process operated in three sequential phases: temporal segmentation of videos, stratified random selection of frames within segments, and validation against multiple heuristic criteria.

Frame validation employed three primary metrics. The first was edge density within designated ROI, particularly in the lower third of the frame, where only candidates exceeding a threshold of 0.15 were accepted. The second metric was temporal stability, which prioritised frames exhibiting less than 5% pixel displacement. The third was visual diversity, requiring a difference greater than 25% from previously selected frames, as determined through histogram distribution analysis.

If no frame met the criteria after 10 sampling attempts, the fallback selected the one with highest edge density. A `tkinter`-based Graphical User Interface (GUI) enabled manual verification and overrides, which proved crucial—around 19% of automatically selected frames were manually replaced due to subtle quality issues missed by the heuristics.

4.1.3 Data Annotation

The extracted frames were then uploaded to the Roboflow platform for systematic annotation. The annotation scheme comprised six distinct classes, each assigned a specific colour for visual differentiation: Breaking News Graphics (Orange), Digital On-Screen Graphics (Red), Lower Third Graphics (Blue), News Headlines (Green), News Tickers (Cyan), and Other News Graphics (Teal).

These classes were selected based on a preliminary analysis of 250 broadcast samples, representing the most common and visually distinctive graphic elements in modern news production. Annotation guidelines (detailed in Appendix B) ensured consistency in bounding box placement, with particular attention to overlapping elements.

In total, **1,500** images yielded **4,749** bounding box annotations, averaging **3.2** annotations per image. The annotated corpus presented an average resolution of **0.92 MP** with a median aspect ratio of **832**$\times$**720px**.

4.1.4 Dataset Analysis & Visualisation

Comprehensive analysis of the NGD revealed key characteristics informing subsequent model design decisions. Class distribution analysis (Figure 4.1) showed Digital On-Screen Graphics as the dominant class (41.9% of annotations), followed by Lower Third Graphics (15.7%) and Other News Graphics (14.0%). This imbalance reflects real-world broadcast

conventions and was preserved to maintain ecological validity.
Spatial distribution analysis through heat maps revealed clear positioning patterns for different graphic types. Lower Third Graphics showed strong localisation in the bottom third of the frame, whilst Breaking News Graphics demonstrated greater positional variability. Moreover, co-occurrence analysis quantified the frequency with which different graphic classes appeared simultaneously. Digital On-Screen Graphics frequently co-existed with Lower Third Graphics and News Tickers, as detailed in the co-occurrence matrix and heat map visualisations provided in the Appendix E.1.

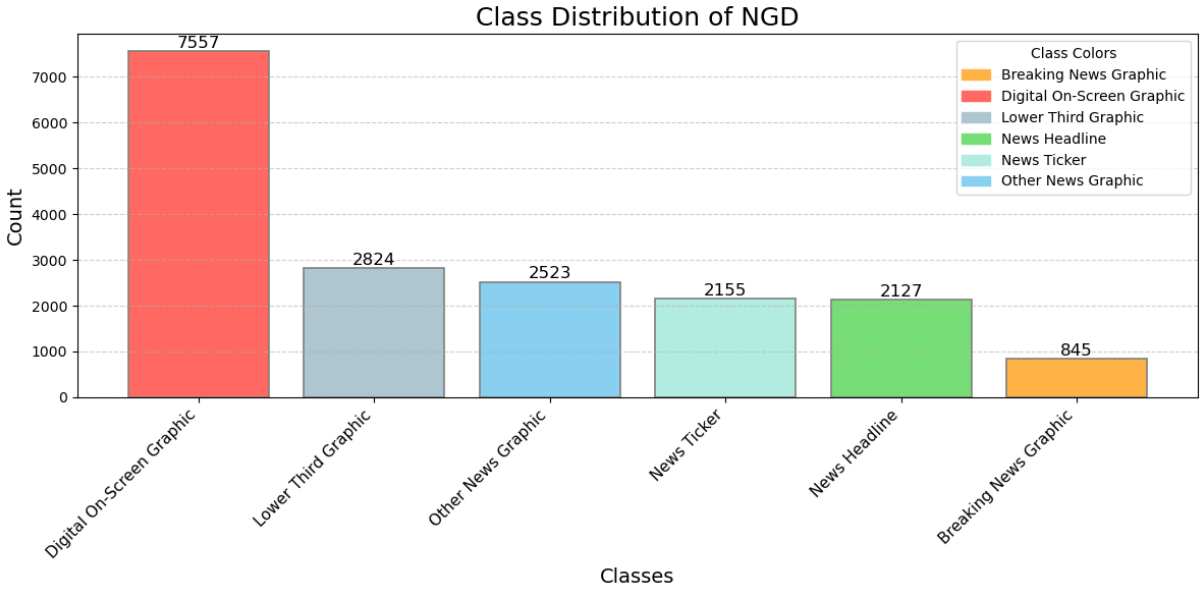


Figure 4.1 Class distribution within the NGD

4.1.5 Data Preprocessing & Augmentation

Several preprocessing strategies were evaluated before model training. The selected approach used **640×640px** resizing with black padding to maintain aspect ratio, automatic orientation correction, and pixel value normalisation to [0,1]. This preserved aspect ratio whilst meeting YOLO's input requirements. Alternative methods like reflection padding and distortion-based resizing were rejected after testing showed degradation in fine text detail.
Controlled data augmentation was applied to enhance model generalisation, with six variants generated per original image. Table 4.1 summarises the augmentation types and parameter ranges used. These values were calibrated through visual inspection to ensure that transformations remained realistic while increasing variability. More aggressive augmentations, such as rotation or horizontal flipping, were deliberately excluded to avoid introducing unnatural graphic orientations.

Augmentation Type	Parameter Range
Random Cropping (Zoom)	0–10%
Brightness Adjustment	±20%
Exposure Variation	±10%
Random Noise Injection	Up to 1.09% of pixels

Table 4.1 Data augmentation settings used during training, calibrated to preserve visual realism while increasing variability

The final dataset was partitioned following a **93%/4%/2%** split (train/validation/test), with stratified sampling ensuring proportional class representation across all partitions.

4.2 Object Detection Model Training YOLO

This section outlines the processes undertaken to train object detection models specialised for news graphic detection using various YOLO architectures.

4.2.1 Roboflow Training

Multiple YOLO architecture variants (YOLOv12, YOLOv11, and YOLO-NAS) were initially trained utilising Roboflow's cloud infrastructure to overcome local hardware limitations. Through comparative testing, the *Fit with Black Edges* resizing technique consistently outperformed alternative approaches, achieving a 2% improvement in mAP. Furthermore, comparative analysis demonstrated the superiority of YOLOv12, which yielded a 2-3% performance increase in detection accuracy when compared to YOLOv11. These findings informed subsequent identification of optimal hyper-parameter configurations.

4.2.2 Local Training

Following cloud-based experimentation, a dedicated local training pipeline was developed on a high-performance workstation (AMD Ryzen 9 7900X3D, 128,GB DDR5 RAM, NVIDIA RTX 4090 Graphics Processing Unit (GPU)). Two specialised Jupyter notebooks were designed: one targeting YOLOv8 training through the Ultralytics framework, and another supporting custom YOLOv12 implementations compatible with Common Objects in Context (COCO) annotation formats.
Multiple local training runs explored hyper-parameter impacts whilst ensuring reproducibility. All models were trained from scratch to exclude pre-trained weights and promote domain-specific learning. The optimal configuration used the `yolo12m.pt` variant with batch size 8, 640px resolution, initial learning rate 0.0001, final learning rate factor

0.00001, dropout rate 0.4, and cosine learning rate scheduling. Early stopping with 10 epochs patience was implemented to mitigate overfitting whilst preserving detection performance.

4.3 The ANEP

The ANEP was developed to reliably extract personal names appearing within graphical overlays in news videos. The system is structured as a comprehensive, multi-stage pipeline integrating modern DL, OCR, and NER techniques. A full flowchart representation of the ANEP is included in Appendix C.4.

4.3.1 Pipeline Overview

The pipeline processes video inputs through five sequential phases:

1. **Frame Processing** – Sampling video frames and applying multi-level deduplication to remove redundant content.
2. **Graphic Detection & Preprocessing** – Detecting graphical overlays using YOLOv12 and applying preprocessing pipelines to the extracted ROIs.
3. **Text Extraction** – Extracting textual information from preprocessed regions using Tesseract OCR.
4. **Named Entity Extraction & Validation** – Identifying and validating personal names through ensemble NER techniques.
5. **Name Clustering & Timeline Generation** – Clustering similar names, selecting canonical representations, and generating appearance timelines.

The ANEP is implemented in Python, leveraging libraries such as `OpenCV`, `Tesseract`, `Ultralytics`, `spaCy`, `transformers`, and `RapidFuzz`. Runtime device optimisation selects between Compute Unified Device Architecture (CUDA), Metal Performance Shaders (MPS), or Central Processing Unit (CPU) processing.
Critical pipeline parameters are summarised in Table 4.2, with values determined through empirical testing to achieve the most reliable balance between performance, accuracy, and computational efficiency across diverse video scenarios.

4.3.2 Frame Processing

Videos are sampled at one FPS by default, with duplicate frame detection utilising the `xxHash` algorithm. Visually identical frames are skipped to reduce computational over-

Parameter	Value
Image Size	640 × 640 pixels
FPS Sampling Rate	1 frame/second
Duplicate Frame Threshold	0.95
Contiguous Skip Threshold	3 frames
YOLO Confidence Threshold	0.6
OCR Confidence Threshold	40
Transformer NER Confidence Threshold	0.85

Table 4.2 Optimised ANEP parameters for efficient frame sampling, redundancy reduction, and high-confidence detection across YOLO, OCR, and NER stages

head. A contiguous similarity threshold prevents redundant processing of near-identical scenes.

4.3.3 Graphic Detection and Preprocessing

Unique frames are analysed using the fine-tuned YOLOv12 object detection model, mentioned in Section 4.2.2. ROI corresponding to news graphics are extracted, and Non-Maximum Suppression (NMS) is applied to remove overlapping detections. Detected ROI are subjected to advanced preprocessing techniques aimed at improving downstream OCR performance, including CLAHE enhancement with Gaussian blurring, Otsu thresholding, adaptive thresholding, greyscale resizing, and morphological clean-ups.

4.3.4 Text Extraction

Preprocessed ROI are passed through Tesseract OCR pipelines, using multiple preprocessing techniques. The highest-confidence result is selected based on a quality metric and stored in structured `.txt` files for subsequent analysis.

4.3.5 Named Entity Extraction & Validation

Extracted text undergoes dual NER analysis using a Transformer-based NER model (fine-tuned `BERT` model) and a spaCy pipeline enhanced with GliNER for zero-shot entity extraction. Detected candidate names are validated through a combination of heuristic rules: minimum word length, absence of numerical characters, appropriate punctuation, and linguistic plausibility checks. Additionally, cross-validation between multiple NER methods further boosts confidence in detected entities.

4.3.6 Name Clustering & Timeline Generation

Recognised names are clustered using fuzzy matching with `RapidFuzz`, Jaccard similarity, and embedding-based cosine similarity using spaCy vectors. Timeline generation includes detailed timestamps of each name's appearance and frequency statistics, all saved in structured JSON outputs. Summary statistics, including the total number of names detected, the number of unique clusters, and the distribution by graphic class, are compiled for analytical evaluation.

4.4 GenAI APIs-Based Video Analysis Pipelines

This section introduces two GenAI-powered pipelines for video frame analysis and name extraction. The first combines the GVA and Gemini 1.5 Pro via Google Cloud Platform[1], while the second uses the LLaMA 4 Maverick LLM via OpenRouter[2]. Both are subject to usage constraints: OpenRouter permits 1,000 requests per day, and Google Cloud access is limited to a $300 initial credit. These limitations informed design choices such as frame sampling rates and retry mechanisms to support efficient and sustainable operation.

4.4.1 Google Cloud Vision & Gemini 1.5 Pro

This system samples video frames at a configurable interval and generates a perceptual hash for each using the DCT. To identify distinct frames, the Hamming distance between successive hashes is computed, retaining only frames whose similarity score falls below a predefined threshold. This deduplication mechanism improves processing efficiency by filtering out visually redundant frames.

Distinct frames are encoded as Joint Photographic Experts Group (JPEG) images and passed to the GVA, which performs `TEXT_DETECTION` on each image. Extracted text snippets across frames are aggregated and passed to the Gemini 1.5 Pro LLM. A prompt template instructs the model to exclusively extract real-world personal names, explicitly excluding brand names, TV channels, show titles, or non-name entities (see Appendix D for the full prompt used).

The system leverages concurrency, using a multi-threaded approach to parallelise frame sampling and text extraction tasks. Summary statistics, including total video frames, distinct frames processed, frames with detected text, names extracted, and processing duration, are compiled and persistently saved.

[1] `https://console.cloud.google.com/`

[2] `https://openrouter.ai/meta-llama/llama-4-maverick:free`

4.4.2 LLaMA 4 Maverick

This pipeline follows a similar overall structure but introduces several enhancements specific to the OpenRouter environment. Distinct frames are identified using a hybrid perceptual hashing approach that combines DCT based hashes with ORB key-point features, improving robustness to subtle visual variations.
Rate limiting is handled through an adaptive mechanism, dynamically adjusting allowed request rates based on recent API responses. Instead of relying on a fixed OCR engine, each frame is sent as a base64-encoded image to the LLaMA model with an explicit text extraction prompt (see Appendix D for the full prompt). A fallback prompt is applied for frames initially returning insufficient text.
Name extraction from the aggregated OCR outputs similarly employs LLaMA 4 Maverick, with a structured JSON format requested directly from the model. Results are saved following the same structure as the Google-based pipeline, enabling comparative evaluation across both approaches.

4.5 ANEP UI

The ANEP's UI was designed as a progressive web application with a step-based workflow architecture. The system's frontend implementation utilises React and TypeScript, supported by Tailwind CSS and Vite. This modern stack was chosen for its modern design capabilities, strong community support, and widespread current adoption, ensuring both developer efficiency and long-term maintainability (see Appendix G.2 to view the GUI).

4.5.1 Architecture & Component Structure

The interface centres on a `VideoAnalyser` component coordinating a five-step workflow. `UploadStep` enables drag-and-drop or manual video file selection, with validation, preview, and metadata extraction. `ModelSelectionStep` lists available extraction models with metrics for speed, accuracy, and API usage. `ConfirmationStep` summarises file and model choices, showing metadata and estimated processing details. `AnalysisStep` manages the processing pipeline with live progress, log output, back-end API calls, and error handling. `ResultsStep` displays extracted names with visualisation tools, and export options.

4.5.2 UX Design

The interface incorporates several UX enhancements to improve usability and accessibility. Progressive visual indicators track workflow completion and provide users with clear navigation cues. The application implements responsive design patterns that adapt to various screen sizes with mobile-specific components to accommodate different usage contexts.

Context-sensitive help and tooltips are integrated throughout the interface to assist users with technical features. The design includes collapsible sections for advanced options and detailed logs, allowing users to access additional information when needed while maintaining a clean primary interface. The application supports both light and dark mode with tailored colour schemes for accessibility.

4.5.3 Dissertation Survey Visualisation

The project incorporates advanced data visualisation capabilities through the Survey-Dashboard component, a custom tool developed to present the survey findings discussed in Section 5.4. The dashboard renders interactive charts and tables from survey data, providing an intuitive interface that allows users to select questions, apply demographic filters, and visualise responses using multiple chart types. Notable features include state persistence between sessions, intelligent categorisation of demographic versus opinion questions, and export capabilities for further analysis.

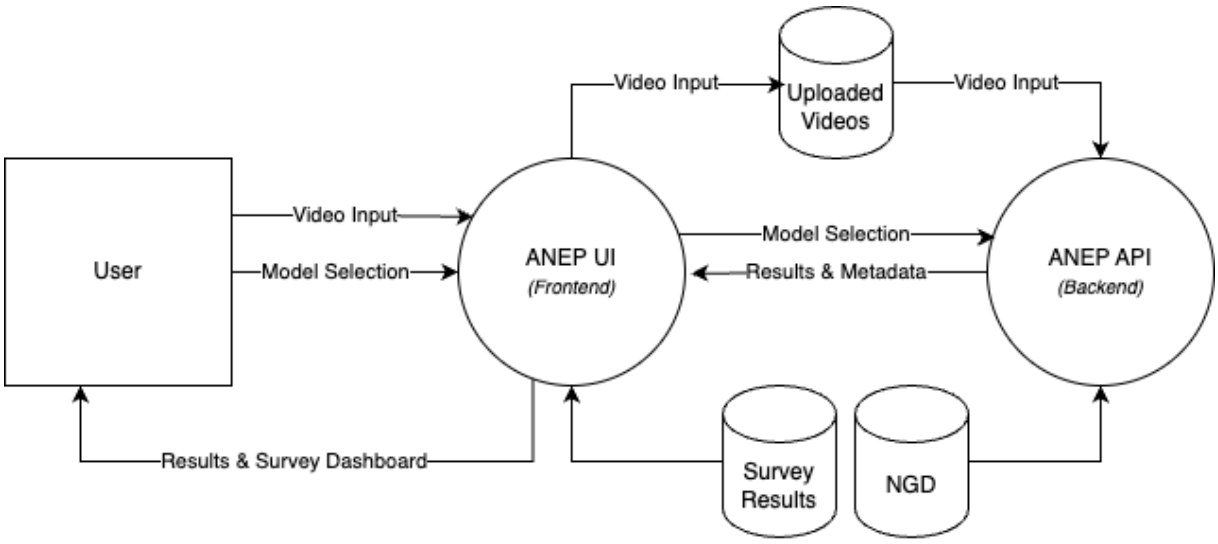


Figure 4.2 Level 0 DFD showing the high-level interaction between the user, ANEP frontend and back-end modules, and associated data stores

5 Evaluation & Results

This chapter evaluates the performance of the proposed system across multiple dimensions. It begins with an in-depth analysis of OD accuracy using the NGD, followed by a comparative assessment of name extraction pipelines and a summary of findings from a user survey.

5.1 Evaluation Metrics

To assess the system's performance across all experimental conditions, several standard quantitative metrics are employed. These metrics provide a comprehensive framework for evaluating detection accuracy, localisation precision, and overall effectiveness of the approach.

5.1.1 Classification Metrics

Precision measures the correctness of positive predictions, capturing what proportion of the objects identified were genuinely correct. Mathematically, it is defined as:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Recall quantifies the system's ability to find all relevant instances, indicating what proportion of the actual objects present were successfully detected. It is calculated as:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

The **F1 Score** offers a balanced evaluation by integrating both precision and recall into a single measure. As the harmonic mean of these two values, it penalises extreme imbalances between precision and recall:

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

5.1.2 Object Detection Metrics

For OD tasks, performance is typically evaluated using variants of the mAP metric, which account for both classification accuracy and bounding box localisation.
mAP@0.5 shows how precise the model is, using a fixed Inersection over Union (IoU) threshold of 0.5. A predicted box counts as correct if it overlaps the ground truth by at

least 50%. This metric balances detection and localisation but may tolerate moderately imprecise boundaries.
mAP@0.5:0.95, by contrast, averages mAP scores across multiple IoU thresholds from 0.5 to 0.95 in 0.05 increments. This stricter evaluation rewards more accurate localisation, providing a more comprehensive measure of detection quality.

5.2 OD Results

This section evaluates YOLO-based models trained on the NGD for news graphic detection, covering model comparison, performance metrics, and visualisations to assess effectiveness.

5.2.1 Model Comparison

As outlined in Section 4.2, several YOLO-based models were trained on the NGD using both Roboflow's cloud and local pipelines. Training ran for up to 150 epochs with early stopping, using a fixed input size of 640×640px. Performance was evaluated using precision, recall, and mAP.
Table 5.1 summarises performance across six representative training runs. While Roboflow based experiments using YOLOv12*(n)* achieved strong results (93.8% mAP@0.5, 91.6% precision, 90.8% recall), the highest overall performance came from the local training run using YOLOv12*(m)* configuration, reaching **95.8% mAP@0.5 with 93.9% precision and 93.5% recall**.
The locally trained YOLOv8*(m)* model achieved 93.7% mAP@0.5 but showed lower recall (86.9%), suggesting detection inconsistency. YOLOv11*(n)* produced competitive results particularly in recall (90.4%), though with slightly lower localisation capability (mAP@0.5: 93.1%). The YOLOv12*(n)* (Reflect) variant showed moderate performance (91.8% mAP@0.5), while YOLO-NAS*(n)* underperformed with 85.1% precision and 84.3% recall, suggesting lower domain generalisability.
The superior performance of the locally trained YOLOv12*(m)* model demonstrates the benefits of increased model capacity and local optimisation. The 95.8% mAP@0.5 represents near-human level accuracy for news graphic detection, suggesting the model is deployment-ready for practical applications.

5.2.2 OD Evaluation Metrics

This evaluation focuses on the `NGD-YOLOv12_v5` *(Local YOLOv12)* model, which achieved the best performance across all OD experiments. Full result graphs for this model are presented in Appendix E.3

Model	Precision	Recall	mAP@0.5	mAP@0.5:0.95	Epochs
Locally Trained Models					
YOLOv12*(m)*	**93.9%**	**93.5%**	**95.8%**	**88.7%**	**102**
YOLOv8*(m)*	92.6%	86.9%	93.7%	75.2%	47
Roboflow Platform Models					
YOLOv12*(n)*	91.6%	90.8%	93.8%	85.4%	120
YOLOv11*(n)*	91.2%	90.4%	93.1%	84.9%	100
YOLOv12*(n)* (Reflect)	91.4%	85.7%	91.8%	80.4%	72
YOLO-NAS*(n)*	85.1%	84.3%	91.0%	61.0%	51

Table 5.1 Comparison of YOLO-based OD models trained on the NGD.
The best-performing model, **YOLOv12*(m)*** was locally trained and led across all metrics.

Ultralytics Evaluation Metrics

Confusion Matrices: Both raw and normalised confusion matrices demonstrated high overall classification accuracy. The most significant challenge emerged between "Digital On-Screen Graphic" and "Background" categories, where 37% of true digital graphics went undetected. This highlights the inherent difficulty in distinguishing ephemeral digital elements from visually similar non-graphic regions, particularly when graphics have subtle boundaries or transparent components.

Precision–Confidence Analysis: The precision–confidence curves revealed distinct performance patterns across categories. "Digital On-Screen Graphic", "News Ticker", and "News Headline" classes consistently achieved excellent precision exceeding 95% at higher confidence thresholds. In contrast, the "Other News Graphic" category showed greater variability, reaching only 88% at its peak. This disparity suggests that whilst well-defined graphic elements are reliably classified, graphics with inconsistent structure or stylistic diversity require more conservative confidence threshold optimisation for deployment.

Recall–Confidence Relationship: High recall was maintained across most categories, with "News Ticker" particularly notable for sustaining nearly complete recall (approaching 100%) even at high confidence thresholds up to 85%. The "Other News Graphic" category again demonstrated comparatively weaker performance, confirming persistent limitations in detecting complex graphics or those spanning multiple screen regions with varied visual characteristics.

Precision–Recall Performance: Precision–recall curves confirmed the model's effectiveness, with five of six classes exceeding 95% mAP@0.5. "Other News Graphic" was the exception at 87.6%. These results highlight strong performance on structured, visually consistent graphics and relative weakness with ambiguous or poorly localised content.

F1 Score Optimisation: Through systematic threshold tuning, a global optimum F1 score of 94% was identified at a confidence threshold of 0.536. This carefully calibrated threshold effectively balances false positives against missed detections, providing an optimal configuration for practical deployment in broadcast monitoring applications.

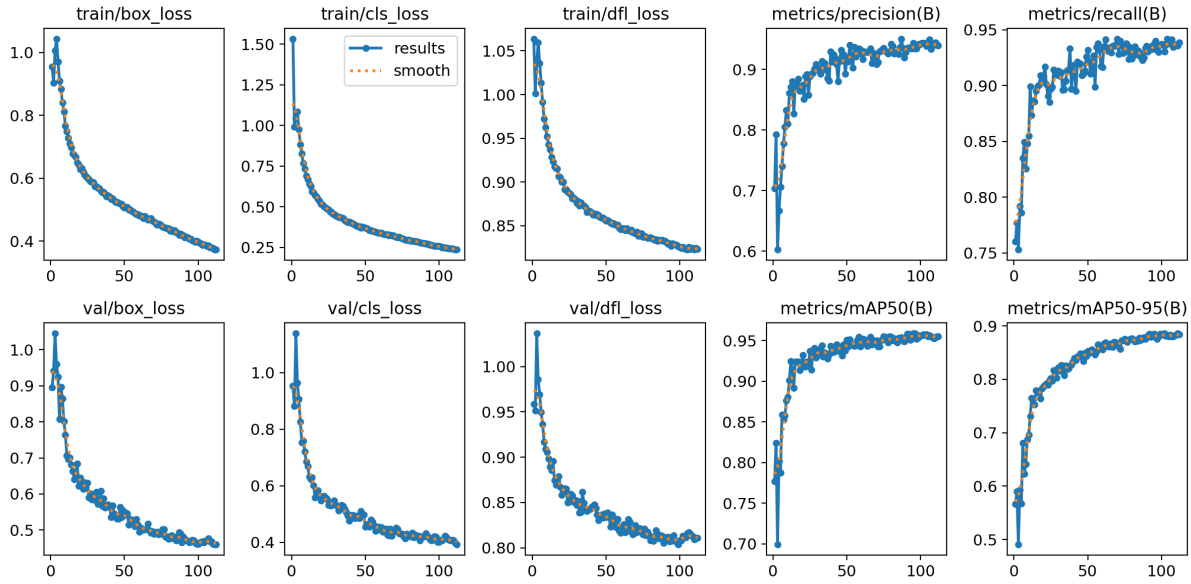


Figure 5.1 Evaluation curves for each class: precision–confidence, recall–confidence, precision–recall, and F1–confidence relationships

Code-Based Metrics and Training Dynamics

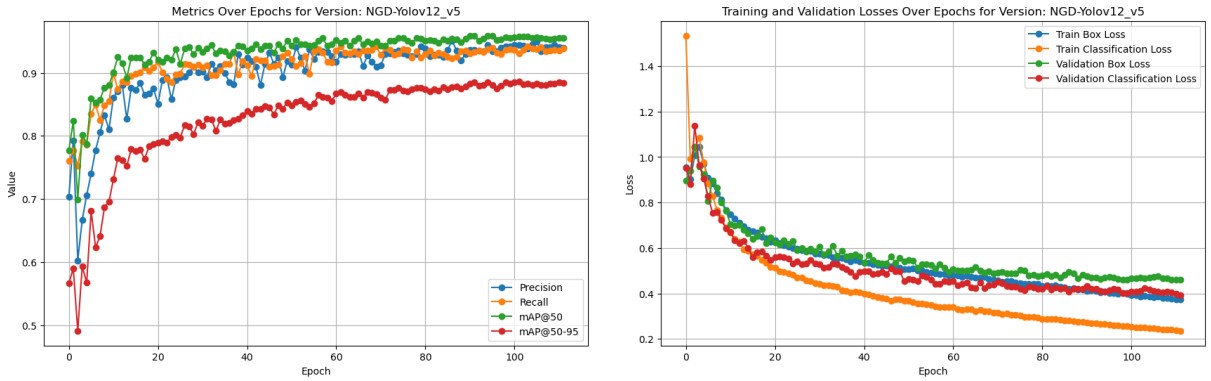
Training and validation metrics extracted from `results.csv` and epoch logs confirmed steady convergence and strong generalisation.

Training Metrics: Figure 5.2a illustrates key metric convergence over 110 epochs. mAP@50 and mAP@50–95 steadily improved before plateauing, whilst precision and recall stabilised above 93%, demonstrating robust learning dynamics and generalisation capability.

Loss Curves: Figure 5.2b shows consistent downward trends across training and validation losses, with final validation box and classification losses reaching 0.46 and 0.39 respectively. The smooth, parallel decline in validation losses indicates minimal overfitting.

Learning Rate Schedule: All three parameter groups were subjected to cosine decay, which stabilised learning and prevented gradient overshoot during later epochs.

Precision–Recall Scatter: Epoch-wise precision-recall pairs clustered distinctly in the upper-right quadrant beyond epoch 70, demonstrating sustained high performance with minimal volatility.



(a) Training progression of precision, recall, mAP@50, and mAP@50–95

(b) Training and validation loss progression across training epochs

Figure 5.2 Training dynamics of NGD-YOLOv12_v5 showing metric convergence and loss over 110 epochs

5.2.3 Model Visualisation

Visual interpretation techniques, including Grad-CAM activation mapping and cross-broadcaster testing, were used to examine the best-performing model's internal representations and generalization capabilities. These visualizations reveal how the model processes news frames and which features drive successful detection, demonstrating robustness across previously unseen sources and formats.

Grad-CAM

Grad-CAM analysis (Figure 5.3) of the YOLOv12 model reveals that early convolutional layers strongly activate around text-dense regions, particularly lower-third graphics and news tickers. This alignment between internal representations and the detection task confirms effective low-level feature encoding for text localisation in broadcast settings.

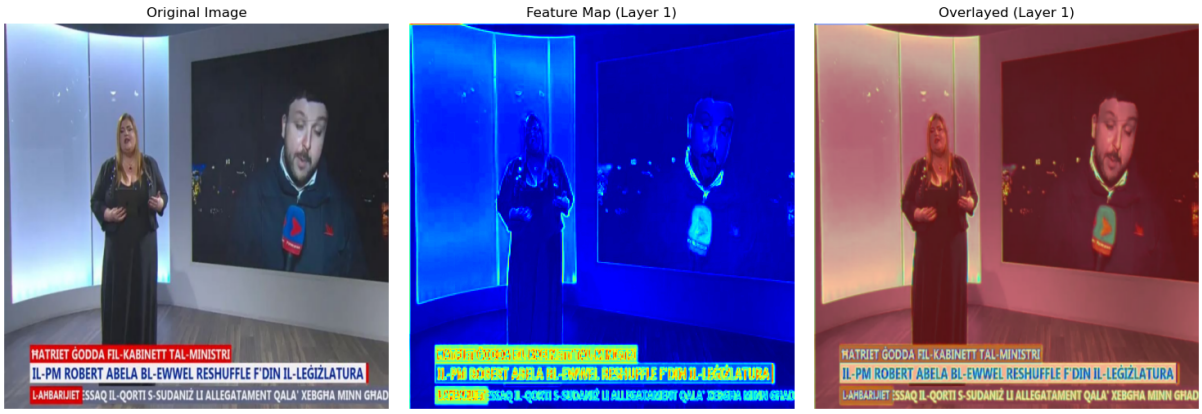


Figure 5.3 Grad-CAM visualisation for a representative news frame, showing original input (left), activation map (centre), and overlay (right) from Layer 1 of YOLOv12

Visualisation Samples

`NGD-YOLOv12_v5` was tested on unseen news broadcasts from diverse sources to assess generalisation beyond the NGD. Bounding boxes identified predicted graphic regions exceeding 0.65 confidence threshold per frame, with supporting visual analytics.
Figure 5.4 shows a local TVM broadcast where the model accurately detected all graphic elements, including overlays in different screen positions.

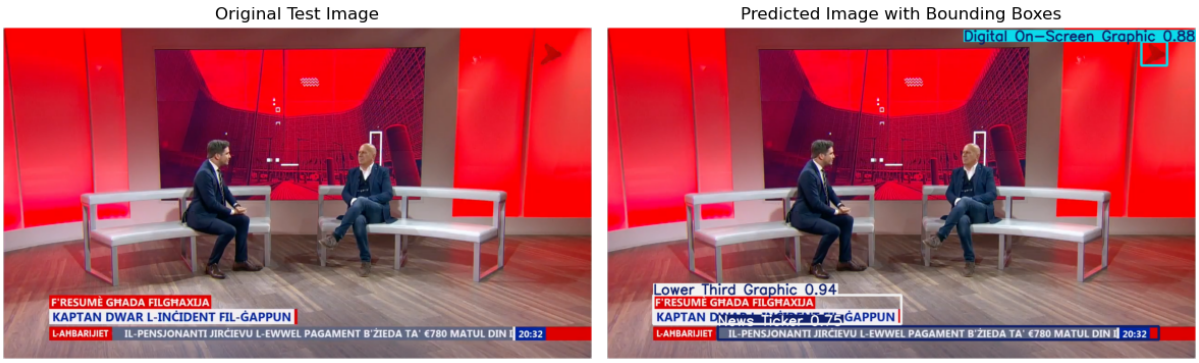


Figure 5.4 Sample 1: Successful detection from TVM news

Figure 5.5 shows a photo of a TV playing Sky News. Despite glare and noise, the model aligned well with one graphic but missed two others, still demonstrating strong performance given the input quality.

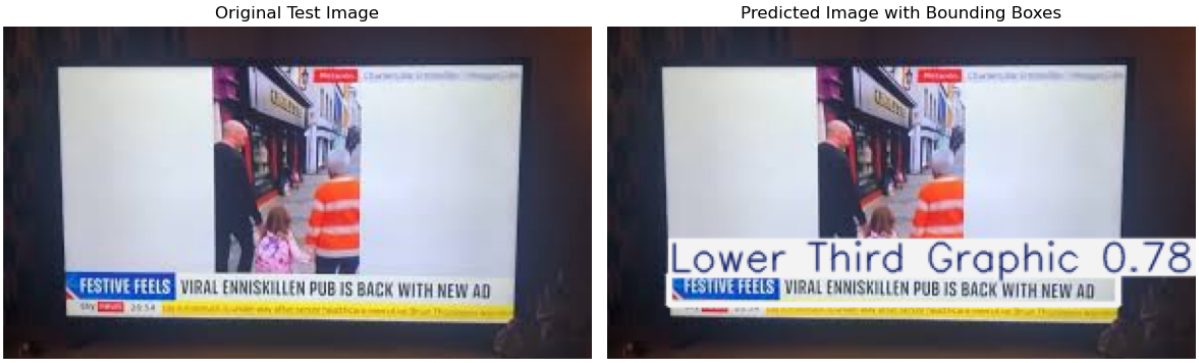


Figure 5.5 Sample 2: Frame extracted from a photo of a Sky News broadcast

Figure 5.6 shows a Rai News broadcast absent from the NGD. The model detected both main graphics but merged ticker with lower third. High confidence and alignment demonstrate strong domain transfer.

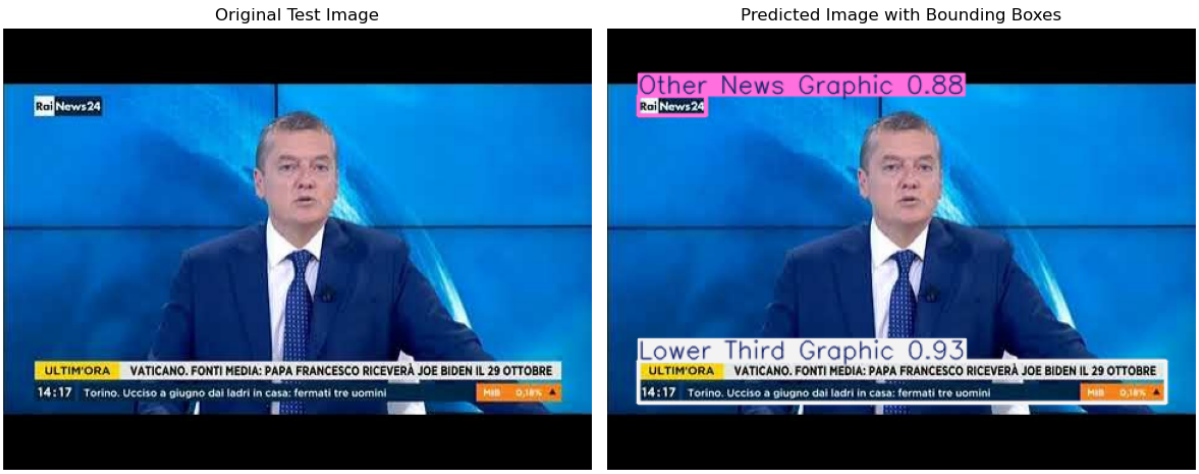


Figure 5.6 Sample 3: Accurate detection from Rai News despite no training exposure

Figure 5.7 shows a Daily Mail social media post. Despite the visual distinctiveness created by TikTok-related overlays, the model precisely identified graphics, demonstrating its ability to generalise to social media-based formats.

Figure 5.7 Sample 4: Generalisation to social media overlay from Daily Mail

The model's strong performance across unseen broadcasters and formats demonstrates robust domain transfer, crucial for real-world deployment where training on every possible source is impractical. Even partial detections in challenging conditions (e.g., photographed screens) suggest the model's features are sufficiently generalisable for diverse applications.

5.3 Comparative Name Extraction Performance

This section presents a comparative evaluation of the three distinct name extraction pipelines applied to the same video material, ensuring a controlled evaluation environment.

5.3.1 Evaluation Methodology and Metrics

Evaluation utilised a manually constructed ground truth set comprising all named individuals appearing in the test video, along with timestamps of their first and last appearances. This annotation process was conducted by a human evaluator who reviewed the video frame-by-frame, recording each name shown in the visual graphics and its associated temporal span. This approach enabled both content-based and time-based comparisons, supporting precise characterisation of each system's ability to detect, identify, and temporally localise names.
Performance was assessed using standard information retrieval metrics: precision, recall and F1 score. These were computed by aligning system outputs with the ground truth list, using case-insensitive string matching and fuzzy tolerance for minor OCR artefacts. A name was considered correctly identified if it appeared within the correct time frame and matched an entry in the annotated set. Total execution time was also recorded to assess runtime efficiency.

Model	Precision (%)	Recall (%)	F1 Score (%)	Time (s)
ANEP	79.92	74.44	68.10	542.15
GVA + Gemini 1.5	**93.33**	**76.67**	**82.22**	**94.68**
LLaMA 4 Maverick	66.67	50.00	55.56	140.18

Table 5.2 Performance of name extraction models in terms of precision, recall, F1 score, and processing time

5.3.2 Quantitative Results and Discussion

As shown in Table 5.2, the **GVA + Gemini 1.5** pipeline achieved the best overall performance across all metrics. With a precision of 93.33% and an F1 score of 82.22%, it outperformed both ANEP and the LLaMA 4 approach. Its high precision indicates that most detected names were correct and relevant to the video context, while its recall suggests successful identification of the majority of ground truth names. This pipeline achieved these results with a relatively low processing time of 94.68 seconds, demonstrating its efficiency and potential for near real-time deployment.

The **ANEP** pipeline showed more balanced precision and recall (72.92% and 74.44% respectively), but a lower F1 score (68.10%), reflecting occasional struggles with consistent name matching across temporal segments. Its lengthy runtime (542.15 seconds) stems from its multi-stage architecture, involving fine-grained OCR, deep NER processing and name clustering logic. Nevertheless, ANEP's modular design allows for explainable intermediate outputs, which may be preferred in critical or forensic contexts where traceability is essential. Despite slower processing, ANEP's modular architecture provides valuable explainability for applications requiring audit trails or manual verification, such as legal or journalistic contexts.
The **LLaMA 4 Maverick** pipeline exhibited the weakest performance, with only 50.00% recall and a low F1 score of 55.56%. This reflects the difficulty of applying LLMs to noisy OCR outputs without structured post-processing or temporal filtering. Despite completing inference in a relatively fast 140.18 seconds, the model frequently misidentified entities or failed to disambiguate between names shown in overlapping or complex visual scenes. This result highlights the challenges of using general-purpose language models in specialised visual domains without additional grounding mechanisms.

5.3.3 Error Analysis and Limitations

While each pipeline demonstrated specific strengths, closer inspection of system outputs reveals recurrent errors and limitations affecting extraction accuracy, generalisability, and temporal alignment. These challenges are particularly relevant when considering the deployment of automated name extraction systems in real-world, multilingual, and noisy broadcast contexts.

NER Errors in ANEP

A prominent source of error within the ANEP pipeline was the misclassification or omission of proper names during the NER stage. Despite achieving relatively balanced recall and precision, several issues emerged in handling contextually dependent or ambiguous names. Terms such as *"Swift"* (in reference to *Taylor Swift*) or *"Trump"* (in reference to *President Donald Trump*) were inconsistently recognised as personal names. These failures typically occurred when names appeared without preceding forenames or used stylistic typography (scenarios where contextual cues are necessary to disambiguate between surnames, verbs, or nouns).
The system occasionally hallucinated entities, erroneously identifying non-name tokens as personal names, particularly with rare words, uncommon surnames, or noisy OCR output. These errors were most prevalent in visually cluttered scenes or when background elements were misinterpreted as part of the graphic overlay.

The system also struggled with accented characters and names of Maltese origin. Names such as *Ċensu*, *Zahra, or Ġużeppi* were inconsistently detected, often due to limitations in the OCR model's handling of diacritics and insufficient representation of Maltese names in the underlying NER training corpus. Compared to the GVA + Gemini pipeline, ANEP's performance on localised or linguistically non-standard names was noticeably weaker.

Alias Resolution Challenges

ANEP's alias resolution mechanism, which merges visually distinct but semantically equivalent names, showed inconsistent performance. It successfully grouped entities like *"Donald Trump"* and *"President Trump"* under one timeline, yet struggled with variants containing different prefixes, OCR errors or minor spelling variations, such as *"Mickey Rourke"* vs *"Brother Rourke"* or *"Jameel Shariff"* vs *"Jameel Shari"*.
These mismatches typically resulted from variations in naming conventions (e.g., titles like *"Brother"*), OCR artefacts, or insufficient semantic similarity—especially where fuzzy or phonetic matching was needed. While the alias resolution reduced straightforward duplication, its effectiveness declined with higher intra-class variability, particularly for infrequent or non-canonical name forms.

Limitations of the LLaMA 4 Pipeline

The LLaMA 4 Maverick pipeline suffered from a critical architectural constraint: its limited context window. While the model could process short sequences of extracted text with some success, its performance deteriorated significantly when applied to longer videos *(beyond approximately five minutes)*. Beyond this threshold, the API context limitations led to truncated inputs, loss of entity tracking, or outright failure to return valid results.
This constraint rendered the LLaMA 4 pipeline unsuitable for continuous name tracking across extended video segments and severely limited its applicability in practical media analysis scenarios. Furthermore, unlike Gemini, LLaMA 4 required carefully curated, pre-tokenised input batches to avoid overflow, thus introducing additional engineering complexity and reducing its robustness in dynamic processing environments.

General Observations

Across all pipelines, errors were more frequent in scenes involving overlapping graphics, rapid transitions or stylised text layouts. In such cases, OCR noise propagated downstream, resulting in either missed entities or False Positive (FP). The lack of cross-modal grounding (e.g., audio or facial identity cues) further limited each pipeline's capacity to resolve ambiguity in name references.

5.4 Survey Analysis

A survey of **404 participants** was conducted to contextualise system design choices within real-world media consumption behaviours. These responses informed key assumptions about news consumption patterns, visibility of names in video graphics, and perceived value of automated name extraction tools. The survey ran between **10 April and 10 May 2025**, with all responses collected anonymously. No Internet Protocol (IP) addresses were tracked, no names were collected, and participants explicitly confirmed consent.

5.4.1 Demographics and News Consumption Patterns

The sample comprised primarily respondents aged 18–24 (37.4%) and 45–54 (27%), with relatively balanced gender distribution (52% male, 47.5% female). With 92.1% of respondents based in Malta, this reinforced the relevance of including local broadcasters in the NGD.
Significantly, 85.9% of respondents reported following news at least once per week. As shown in Figure 5.8, social media emerged as the dominant news source (79.5%), surpassing both online news websites (53%) and traditional television broadcasts (38.4%). Meta-owned platforms, primarily Facebook (66.9%), Instagram (51.2%), and WhatsApp/Messenger (17.4%), accounted for a substantial proportion of news exposure, highlighting Meta's central role in news distribution, particularly amongst younger audiences.

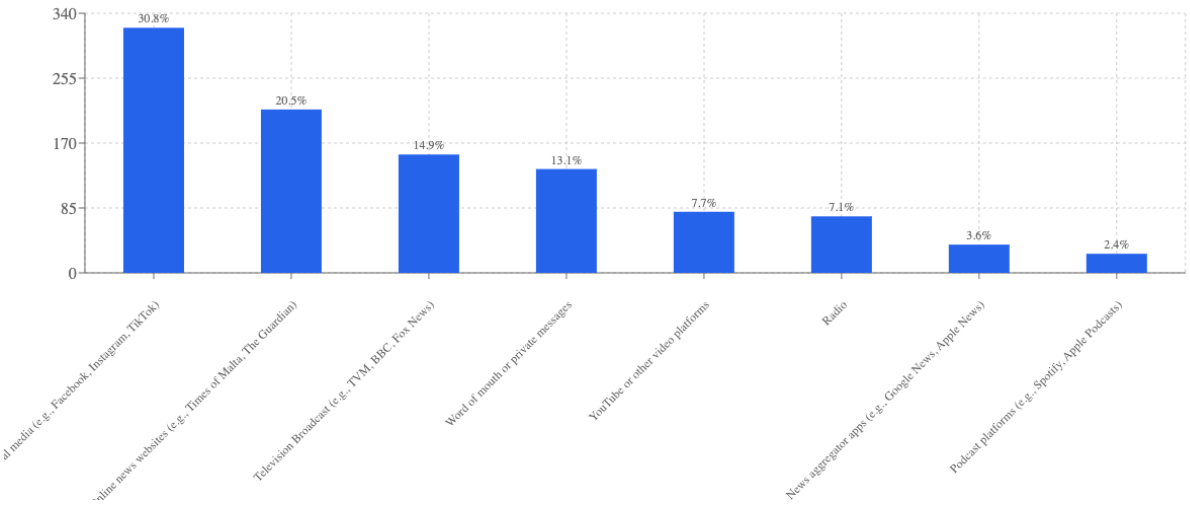


Figure 5.8 Preferred sources for accessing news content

A clear majority (74%) expressed preference for short video clips over full-length news reports (19.6%), as shown in Figure 5.9. Text-based articles (48.3%) and image-based content (44.8%) also remained popular, whilst formats such as infographics (28%) and

podcasts (12.4%) attracted smaller interest. These findings support the decision to focus the ANEP on short-form video content, reflecting real-world consumption habits whilst enabling efficient analysis of news graphics and named entities.

Despite the shift towards social and short-form media, trust in these platforms was comparatively lower. On a five-point scale, traditional television received the highest average trust rating (3.51), followed by online news websites (3.33). Social media platforms scored lower (2.72), with YouTube and podcasts slightly behind (2.66). For television specifically, 53% of respondents rated their trust as 4 or 5, compared to only 18.4% for social media. These results motivated the design of a pipeline to extract names directly from verified visual content rather than relying on potentially unreliable user commentary or external metadata.

When asked whether they fact-check news or names before believing or sharing content, 75.6% of respondents indicated they *always* or *sometimes* do so. The remaining 24.4% who answered *rarely*, *never*, or *not sure* highlight an important gap addressable through reliable, automated systems. This reinforces the relevance of the proposed approach for supporting timely, trustworthy verification.

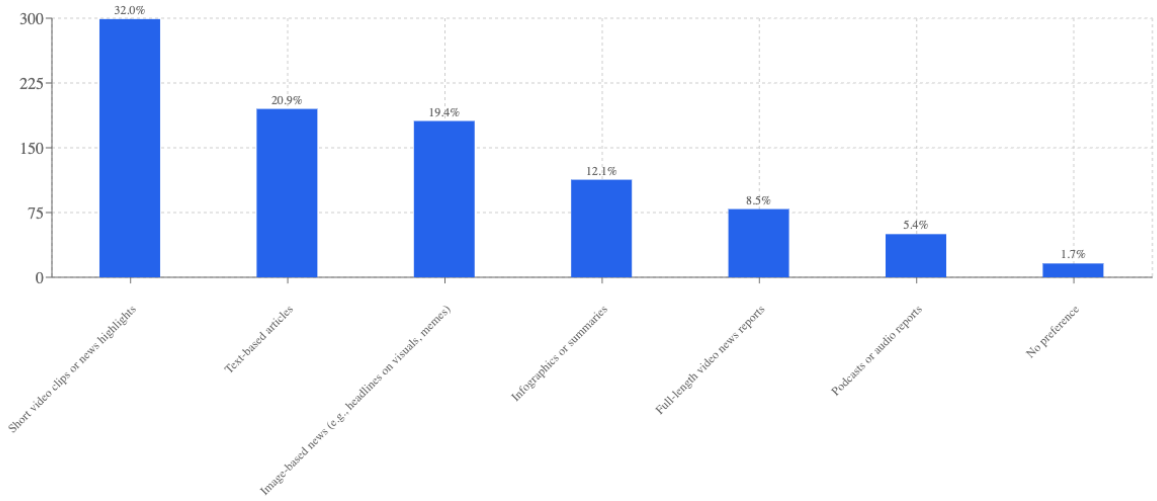


Figure 5.9 Preferred formats for news consumption

5.4.2 Name Graphics and Viewer Interaction

The survey revealed that 46% of respondents occasionally notice names displayed in news video graphics, with 30.2% doing so frequently. Moreover, 82.1% rely on these visual cues to identify individuals or understand their role in news stories.

Accessibility issues were evident: 59.7% reported difficulty reading names in news graphics due to formatting, speed, or clarity (7.7% frequently, 52% sometimes) as shown in Figure 5.10. Additionally, 58.2% had paused or rewound a video specifically to double-check a person's name (15.2% multiple times, 43% once or twice). A further 10% had

never done so but wanted to. These behaviours underscore the importance of extractable name displays, motivating the development of ANEP.
Regarding the necessity of identifying individuals in news videos, 56.4% believed this should be done only when relevant to the story, while 32.2% felt it should always be done. Only 8.9% considered it unnecessary, demonstrating strong support for comprehensive identification practices.

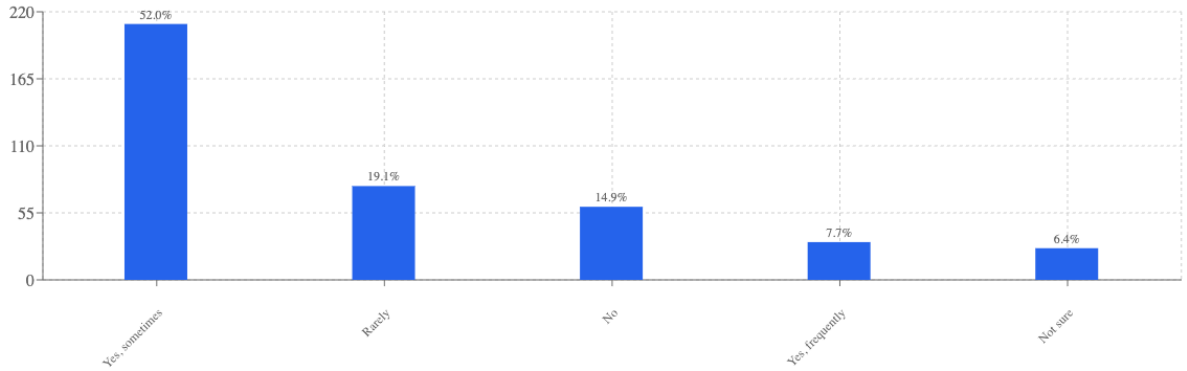


Figure 5.10 Respondents who reported difficulty reading names in news video graphics

5.4.3 Perceptions of Automated Tools

When surveyed about the utility of an automated name extraction tool, 58.3% answered *Yes*, 34.2% *Maybe*, and merely 7.5% *No*. As illustrated in Figure 5.11, 62.6% indicated that trust *depends on who developed it or how it works*, 20.8% expressed confidence, 11.9% lacked certainty, and just 4.7% reported distrust. These findings reveal widespread interest moderated by concerns regarding reliability and transparency, directly shaping the pipeline's emphasis on precision, explainability, and user-centred functionality.

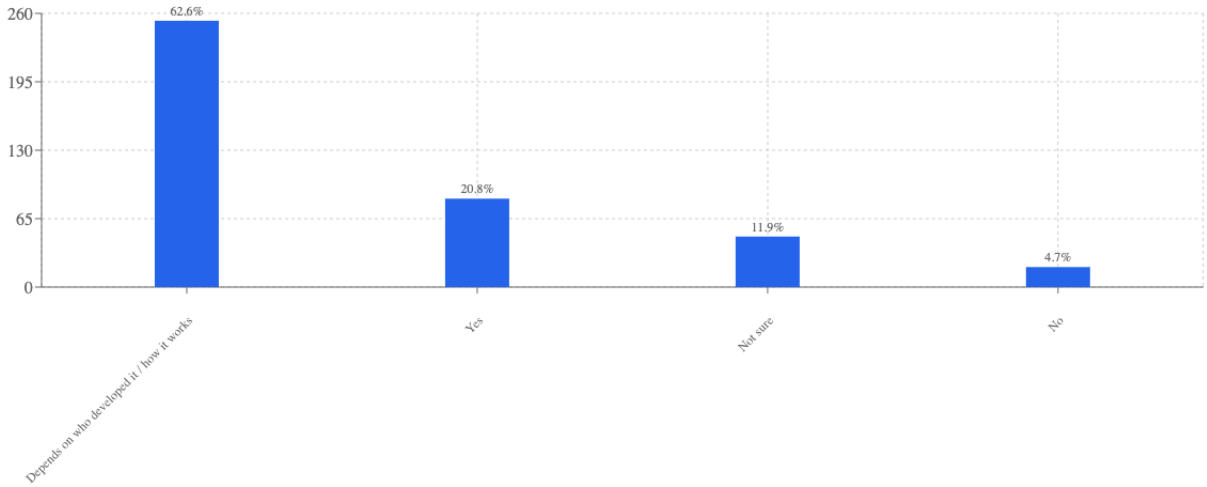


Figure 5.11 Trust in AI-based name extraction tools among participants

6 Conclusion

This chapter presents a comprehensive summary of the dissertation's contributions to automated name extraction from news video graphical overlays. It summarises the research outcomes, including the NGD, the ANEP, comparative evaluations with GenAI approaches, quantitative findings, methodological limitations, and prospective research directions for advancing multimodal video content analysis.

6.1 Summary of Contributions

This dissertation presents a comprehensive investigation into automated extraction of personal names from graphical overlays in news video content. The research addresses a technically challenging and previously under-explored problem at the intersection of CV, OCR, and NLP, with particular emphasis on NER for validating and structuring extracted text. The principal contributions of this work are:

Firstly, the introduction of the NGD, a novel and purpose-built dataset comprising annotated broadcast and social media news frames. This carefully curated dataset captures the stylistic diversity of graphical overlays across multiple sources, including traditional television broadcasts and short-form social media formats. It contains detailed annotations for six distinct classes of news graphics and has been made publicly available to support future research.

Secondly, the proposal and implementation of the ANEP, an end-to-end system integrating a fine-tuned YOLOv12 model for OD, domain-specific OCR pipelines, and a dual-layer NER framework combining spaCy with GLINER and a transformer-based entity recognition model. The pipeline incorporates advanced frame sampling, deduplication, and preprocessing routines to ensure robustness against visual noise, animation, and temporal instability characteristic of broadcast media.

Thirdly, a rigorous comparative evaluation between the ANEP and two GenAI-based name extraction pipelines powered by GVA and Gemini, as well as LLaMA 4 Maverick. This comparative study employed quantitative metrics including precision, recall, F1 score, and processing time to examine the relative strengths of traditional and generative approaches, contributing to discourse on operational trade-offs between deterministic pipelines and multimodal foundation models.

Additionally, the development of a fully functional, user-centred web interface to operationalise the proposed methodologies, supporting video upload, model selection, real-time progress monitoring, and results visualisation, with an integrated interactive survey dashboard.

Finally, the incorporation of a user-centred survey exploring audience engagement with

name graphics in news video content. The survey responses yielded valuable insights into user preferences, perceived challenges, and expectations, informing the NGD composition and substantiating the broader relevance of automated name extraction tools in contemporary media consumption.
These contributions advance automated analysis of news video graphics, demonstrating how carefully designed OD-OCR-NER pipelines, informed by user research and augmented by comparative evaluation with GenAI models, can yield accurate and operationally viable solutions for media indexing, archival search, and accessibility enhancement.

6.2 Key Findings and Results

The comparative evaluation revealed that the GVA and Gemini pipeline achieved the highest name extraction performance with an F1 score of 82.22%, precision of 93.33%, and recall of 76.67%. It demonstrated rapid processing time, supporting potential near real-time deployment. The ANEP pipeline produced balanced precision and recall but a lower F1 score of 68.10%, due to variability in OCR and NER consistency across frames. The LLaMA pipeline achieved an F1 score of 55.56%, hampered by context window limitations and poor robustness to noisy inputs.
The YOLO detection model exhibited strong generalisability to unseen content, accurately localising graphic regions across diverse formats including social media overlays, international broadcasts, and low-quality footage. Despite some class-level confusion, particularly for "Other News Graphic", the best-performing model achieved 95.8% mAP@-0.5 and maintained reliable detection under stylistic variation and visual noise.
Survey responses from 404 participants highlighted significant challenges with name visibility: 59.7% reported difficulties reading name graphics, and 58.2% had paused or rewound videos to identify individuals. Moreover, 82.1% relied on visual name cues for understanding news content, and 92.5% expressed interest in automated extraction tools. These findings validate the practical importance of the problem and support the system's design goals.

6.3 Limitations

Despite the encouraging results, several limitations merit acknowledgement. Firstly, while diverse in source and format, the NGD may not fully capture the breadth of graphical styles across all international broadcasters. Specific design conventions, regional typographies, and non-Western scripts remain under-represented, potentially limiting model generalisability in global applications.

Secondly, although the YOLOv12 OD achieved high mAP scores and generalised well to unseen content, its performance occasionally degraded under conditions of extreme compression, motion blur, or partial occlusion—scenarios common in low-bandwidth or live-streamed news content.
Thirdly, the OCR process remains sensitive to artefacts introduced by animated overlays and anti-aliased fonts. Even with extensive preprocessing and multiple inference passes, Tesseract exhibited inconsistent recognition quality on thin or stylised text, affecting downstream NER performance.
Finally, reliance on external APIs such as GVA, Gemini, and LLaMA introduced constraints related to rate limiting, latency, and cost. The free-tier access used during development imposed sampling restrictions that limited temporal granularity in the GenAI pipelines, potentially affecting comparative fairness in performance benchmarking.

6.4 Future Work

Several promising avenues exist to extend this research. Expanding the NGD to include multilingual broadcast content, particularly with right-to-left scripts or non-Latin alphabets, would improve cross-linguistic generalisability. This could include annotation of speaker labels, affiliation indicators, and temporal persistence metadata to support richer downstream tasks such as role attribution or dialogue summarisation [90].
Another direction involves developing end-to-end multimodal name extraction systems. Emerging transformer-based architectures that jointly model visual, textual, and auditory modalities, such as vision-language LLMs with integrated ASR, offer potential to resolve identity ambiguities by correlating spoken names with visual overlays in real time [91, 92].
Additionally, future implementations could explore optimised, low-latency variants of the ANEP pipeline for deployment in live broadcast environments through model pruning, quantisation, or edge deployment strategies to reduce computational overhead [93, 94].
Finally, system robustness could be enhanced through confidence-aware ensembles, cross-modal validation signals, and self-supervised learning from temporally consistent name mentions, enabling effective operation across unlabelled corpora and facilitating scalable name discovery in large video archives [95, 96].

References

[1] P. Aussu, "Information overload: Coping mechanisms and tools impact", in *Lecture Notes in Business Information Processing*, 2023, pp. 661–669. DOI: 10.1007/978-3-031-33080-3_49.

[2] P. G. Roetzel, "Information overload in the information age: A review of the literature from business administration, business psychology, and related disciplines with a bibliometric approach and framework development", *BuR - Business Research*, vol. 12, no. 2, pp. 479–522, Jul. 2018. DOI: 10.1007/s40685-018-0069-z.

[3] A. A. Naz and R. A. Akbar, "Use of media for effective instruction and its importance: Some considerations", *Journal of Elementary Education*, vol. 18, no. 1–2, pp. 35–40, 2008.

[4] K. Anoop, M. P. Gangan, and V. L. Lajish, "Mathematical morphology and region clustering based text information extraction from malayalam news videos", in *Advances in Intelligent Systems and Computing*, 2015, pp. 431–442. DOI: 10.1007/978-3-319-28658-7_37.

[5] M. D. A. Asif, U. U. Tariq, M. N. Baig, and W. Ahmad, "A novel hybrid method for text detection and extraction from news videos", *Middle-East Journal of Scientific Research*, vol. 19, pp. 716–722, 2014, ISSN: 1990-9233. DOI: 10.5829/idosi.mejsr.2014.19.5.21019.

[6] X. Ma, "Research on short news video transmission in the fusion media environment", *Probe - Media and Communication Studies*, vol. 2, no. 2, p. 25, Feb. 2020. DOI: 10.18686/mcs.v2i2.1295.

[7] C. Zachlod, O. Samuel, A. Ochsner, and S. Werthmüller, "Analytics of social media data – state of characteristics and application", *Journal of Business Research*, vol. 144, pp. 1064–1076, 2022, ISSN: 0148-2963. DOI: 10.1016/j.jbusres.2022.02.016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0148296322001321.

[8] K. Choroś, "Video structure analysis and content-based indexing in the automatic video indexer avi", in *Advances in Multimedia and Network Information System Technologies*, Springer, 2010, pp. 79–90.

[9] S. Lee and K. Jo, "Strategy for automatic person indexing and retrieval system in news interview video sequences", in *2017 10th International Conference on Human System Interactions (HSI)*, IEEE, 2017, pp. 212–215.

[10] H. Zhang and Y. Gong, “Automatic parsing of news video”, in *Proceedings of the IEEE International Conference on Multimedia Computing and Systems*, Boston, MA, USA, 1994, pp. 45–54.

[11] M. Ghoniem, D. Luo, J. Yang, and W. Ribarsky, “Newslab: Exploratory broadcast news video analysis”, in *Proceedings of IEEE VAST*, 2007, pp. 123–130. DOI: 10.1109/vast.2007.4389005.

[12] M. S. Pattichis, V. Jatla, and A. E. U. Cerna, *A review of machine learning methods applied to video analysis systems*, 2023. arXiv: 2312.05352 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2312.05352.

[13] O. Järvi, “News graphics: Some typological and textual aspects”, *LSP & Professional Communication*, vol. 2, no. 2, pp. 8–22, Oct. 2002, ISSN: 1601-1929.

[14] J. R. Fox, A. Lang, Y. Chung, S. Lee, N. Schwartz, and D. Potter, “Picture this: Effects of graphics on the processing of television news”, *Journal of Broadcasting & Electronic Media*, vol. 48, no. 4, pp. 646–674, Dec. 2004. DOI: 10.1207/s15506878jobem4804_7.

[15] D. Seychell, G. Hili, J. Attard, and K. Makantatis, “Ai as a tool for fair journalism: Case studies from malta”, in *2024 IEEE Conference on Artificial Intelligence (CAI)*, 2024, pp. 127–132. DOI: 10.1109/CAI59869.2024.00032.

[16] J. Attard and D. Seychell, *Comparative analysis of image, video, and audio classifiers for automated news video segmentation*, 2025. arXiv: 2503.21848 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2503.21848.

[17] J. S. Foote and A. C. Saunders, “Graphic forms in network television news”, *Journalism Quarterly*, vol. 67, no. 3, pp. 501–507, 1990. DOI: 10.1177/107769909006700304.

[18] L. Buijs, Y. de Haan, and G. Smit, “Using information visualization in the media”, in *Conference Papers*, 2013. [Online]. Available: https://www.internationalhu.com/research/publications/using-information-visualization-in-the-media.

[19] S. Li and P. Jongbin, “The impact of social media on visual communication design”, *Journal of New Media and Economics*, vol. 1, no. 2, pp. 138–145, Mar. 2024. DOI: 10.62517/jnme.202410223.

[20] R. Borgo *et al.*, “A Survey on Video-based Graphics and Video Visualization”, in *Eurographics 2011 - State of the Art Reports*, N. John and B. Wyvill, Eds., The Eurographics Association, 2011. DOI: 10.2312/EG2011/stars/001-023.

[21] N. H. Li, D. Doermann, and O. Kia, “Automatic text detection and tracking in digital video”, vol. 9, no. 1, pp. 147–156, Jan. 2000. DOI: 10.1109/83.817607.

[22] I. Aljarrah and D. Mohammad, "Video content analysis using convolutional neural networks", vol. 1, 2018, pp. 122–126. DOI: 10.1109/iacs.2018.8355453.

[23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788. DOI: 10.1109/cvpr.2016.91.

[24] G. Lavanya and S. D. Pande, "Enhancing real-time object detection with yolo algorithm", *EAI Endorsed Transactions on Internet of Things*, vol. 10, Dec. 2023. DOI: 10.4108/eetiot.4541.

[25] Y. Zhao and S. Wang, "Research on real-time object detection based on yolo algorithm", *Highlights in Science Engineering and Technology*, vol. 7, pp. 323–331, Aug. 2022. DOI: 10.54097/hset.v7i.1091.

[26] S. Gothane, "A practice for object detection using yolo algorithm", *International Journal of Scientific Research in Computer Science Engineering and Information Technology*, pp. 268–272, Apr. 2021. DOI: 10.32628/cseit217249.

[27] J. Terven, D.-M. Córdova-Esparza, and J.-A. Romero-González, "A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas", *Machine Learning and Knowledge Extraction*, vol. 5, no. 4, pp. 1680–1716, 2023. DOI: 10.3390/make5040083.

[28] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning", *Journal of Big Data*, vol. 6, no. 1, Jul. 2019. DOI: 10.1186/s40537-019-0197-0.

[29] A. Qin, M. Xiao, B. Huang, and X. Zhang, "Maze: A cost-efficient video deduplication system at web-scale", in *Proceedings of the 30th ACM International Conference on Multimedia*, ser. MM '22, New York, NY, USA: Association for Computing Machinery, 2022, pp. 3163–3172, ISBN: 9781450392037. DOI: 10.1145/3503161.3548145.

[30] N. Lian *et al.*, *Autossvh: Exploring automated frame sampling for efficient self-supervised video hashing*, 2025. arXiv: 2504.03587 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2504.03587.

[31] K. Hamad and M. Kaya, "A detailed analysis of optical character recognition technology", *International Journal of Applied Mathematics Electronics and Computers*, vol. 4, no. Special Issue-1, p. 244, Dec. 2016. DOI: 10.18100/ijamec.270374.

[32] P. Shruthi and C. D. Verma, "A detailed study and recent research on ocr", *Zenodo*, Mar. 2021. DOI: 10.5281/zenodo.4578126. [Online]. Available: https://doi.org/10.5281/zenodo.4578126.

[33] J. Wang, "A study of the ocr development history and directions of development", *Highlights in Science Engineering and Technology*, vol. 72, pp. 409–415, Dec. 2023. DOI: 10.54097/bm665j77.

[34] M. Heidarysafa, J. Reed, K. Kowsari, A. R.Leviton, J. Warren, and D. Brown, "From videos to urls: A multi-browser guide to extract user's behavior with optical character recognition", *Advances in Intelligent Systems and Computing*, Apr. 2019.

[35] B. Mohit, "Named entity recognition", in *Theory and Applications of Natural Language Processing*, 2014, pp. 221–245. DOI: 10.1007/978-3-642-45358-8_7.

[36] M. Munnangi, *A brief history of named entity recognition*, 2024. arXiv: 2411.05057 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2411.05057.

[37] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding", *arXiv preprint arXiv:1810.04805*, 2018. [Online]. Available: https://arxiv.org/pdf/1810.04805.pdf.

[38] E. Arslan, *Natural language processing: Named entity recognition (ner)*, https://medium.com/@erhan_arslan/natural-language-processing-named-entity-recognition-ner-step-4-3c079c878332, Nov. 2024.

[39] K. Nastou, M. Koutrouli, S. Pyysalo, and L. J. Jensen, "Improving dictionary-based named entity recognition with deep learning", *Bioinformatics*, vol. 40, no. Supplement_2, pp. ii45–ii52, Sep. 2024. DOI: 10.1093/bioinformatics/btae402.

[40] T. S. Musalamadugu and H. Kannan, "Generative ai for medical imaging analysis and applications", *Future Medicine AI*, Sep. 2023. DOI: 10.2217/fmai-2023-0004.

[41] *Comparative study on ai and ocr for data extraction*, https://thirdeyedata.ai/comparative-study-on-ai-and-ocr-for-data-extraction/, Accessed: May 03, 2025, 2024.

[42] J. Hong *et al.*, "Analysis of faces in a decade of us cable tv news", in *Proceedings of the 27th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, Virtual Event, Singapore: Association for Computing Machinery, 2021, pp. 3011–3021, ISBN: 9781450383325. DOI: 10.1145/3447548.3467134.

[43] S. H. Lee, J. W. Ahn, and K. H. Jo, "Automatic name line detection for person indexing based on overlay text", *Journal of Multimedia Information System*, vol. 2, no. 1, pp. 163–170, 2015. DOI: 10.9717/jmis.2015.2.1.163.

[44] D. Saravanan, N. Lavanya, and K. Mahesh, "An extraction of overlay text from digital videos", *Research Inventy: International Journal of Engineering And Science*, vol. 5, no. 4, pp. 88–94, Apr. 2014, ISSN: 2278-4721. [Online]. Available: https://www.researchinventy.com/papers/v5i4/0054088094.pdf.

[45] M. A. A. Hammoudeh, M. Alsaykhan, R. Alsalameh, and N. Althwaibi, “Computer vision: A review of detecting objects in videos – challenges and techniques”, *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 18, no. 01, pp. 15–27, 2022. DOI: 10.3991/ijoe.v18i01.27577.

[46] R. Sapkota, Z. Meng, M. Churuvija, X. Du, Z. Ma, and M. Karkee, *Comprehensive performance evaluation of yolov12, yolo11, yolov10, yolov9 and yolov8 on detecting and counting fruitlet in complex orchard environments*, 2025. arXiv: 2407.12040 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2407.12040.

[47] G. Ulutas, B. Ustubioglu, and M. U. et al., “Frame duplication detection based on bow model”, *Multimedia Systems*, vol. 24, pp. 549–567, Oct. 2018, Published online December 15, 2017. DOI: 10.1007/s00530-017-0581-6. [Online]. Available: https://doi.org/10.1007/s00530-017-0581-6.

[48] S. Gupta, M. Kumar, and A. Garg, “Improved object recognition results using sift and orb feature detector”, *Multimedia Tools and Applications*, vol. 78, no. 23, 2019. DOI: 10.1007/s11042-019-08232-6.

[49] G. Preethi, S. Naik, M. Kenchol, S. P. Jakalannanavar, and M. S. Rachana, “Object detection using fasterrcnn, yolov7 & yolov8”, *Indiana Journal of Multidisciplinary Research*, vol. 04, no. 03, pp. 136–141, 2024.

[50] K. P. H and R. B. Venkatapur, “Deep learning technique for object detection from panoramic video frames”, *International Journal of Computer Theory and Engineering*, vol. 14, no. 1, pp. 20–26, 2022. DOI: 10.7763/ijcte.2022.v14.1306.

[51] J. R. et al., *You only look once: Unified, real-time object detection*, 2016. [Online]. Available: https://arxiv.org/pdf/1506.02640.pdf.

[52] T. M. et al., “Performance study of yolov5 and faster r-cnn for autonomous navigation”, 2022.

[53] L. T. R. et al., *A decade of you only look once (yolo) for object detection*, 2025.

[54] Y. Tian, Q. Ye, and D. Doermann, *Yolov12: Attention-centric real-time object detectors*, 2025.

[55] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks”, 2016. arXiv: 1506.01497 [cs.CV]. [Online]. Available: https://arxiv.org/abs/1506.01497.

[56] B. Bhan and S. Patel, “Efficient medical image enhancement using clahe enhancement and wavelet fusion”, *International Journal of Computer Applications*, vol. 167, pp. 1–5, Jun. 2017. DOI: 10.5120/ijca2017913277.

[57] H. Kamwal, *Enhancing image techniques for better ocr extraction*, 2025. [Online]. Available: https://medium.com/@Hitesh.kamwal/enhancing-image-techniques-for-better-ocr-extraction-15-proven-techniques-to-solve-real-world-b52e8655a9d0.

[58] A. Mumuni and F. Mumuni, "Data augmentation: A comprehensive survey of modern approaches", *Array*, 2022. [Online]. Available: https://doi.org/10.1016/j.array.2022.100258.

[59] N. Ashraf, S. Y. Arafat, and M. J. Iqbal, "An analysis of OCR methods", *International Journal of Computational Linguistics Research*, vol. 10, no. 3, pp. 81–91, Sep. 2019. DOI: 10.6025/jcl/2019/10/3/81-91.

[60] R. Smith, "An overview of the tesseract ocr engine", in *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, vol. 2, 2007, pp. 629–633. DOI: 10.1109/ICDAR.2007.4376991.

[61] P. X. Nguyen, K. Wang, and S. Belongie, "Video text detection and recognition: Dataset and benchmark", in *IEEE Winter Conference on Applications of Computer Vision*, 2014, pp. 776–783. DOI: 10.1109/WACV.2014.6836024.

[62] E. Zacharias, M. Teuchler, and B. Bernier, "Image processing based scene-text detection and recognition with tesseract", 2020. arXiv: 2004.08079 [cs.CV]. [Online]. Available: https://arxiv.org/abs/2004.08079.

[63] R. Kannao and P. Guha, *Overlay text extraction from TV news broadcast*, https://arxiv.org/abs/1604.00470v1, Apr. 2016.

[64] H. Byun, I. Jang, and Y.-W. Choi, "Text extraction in digital news video using morphology", pp. 341–352, Aug. 2002. DOI: 10.1007/3-540-45869-7_39.

[65] M. Li *et al.*, "Trocr: Transformer-based optical character recognition with pre-trained models", 2022. arXiv: 2109.10282 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2109.10282.

[66] M. Muthusundari, A. Velpoorani, S. V. Kusuma, T. L., and O. K. Rohini, "OCR system using AI", *LatIA*, vol. 2, p. 98, Aug. 2024. DOI: 10.62486/latia202498.

[67] B. K. Pattanayak, A. K. Biswal, S. R. Laha, S. Pattnaik, B. B. Dash, and S. S. Patra, "A novel technique for handwritten text recognition using easy OCR", *2023 International Conference on Smart Systems for Applications and Services (SSAS)*, pp. 1115–1119, 2023. DOI: 10.1109/icssas57918.2023.10331704.

[68] K. Smelyakov, A. Chupryna, D. Darahan, and S. Midina, "Effectiveness of modern text recognition solutions and tools for common data sources", *International Journal of Computer Applications*, 2021.

[69] S. Li and Y. Liu, "News video title extraction algorithm based on deep learning", *IEEE Access*, vol. 9, pp. 12 143–12 157, 2021. DOI: 10.1109/ACCESS.2021.3051613.

[70] N. Bhojne, "News video indexing and retrieval using overlay text", in *Computer Science & Information Technology (CS&IT)*, 2012, pp. 11–17. DOI: 10.5121/csit.2012.2302.

[71] M. Marrero, J. Urbano, S. Sánchez-Cuadrado, J. Morato, and J. M. Gómez-Berbís, "Named entity recognition: Fallacies, challenges and opportunities", *Computer Standards & Interfaces*, vol. 35, no. 5, pp. 482–489, 2013, ISSN: 0920-5489. DOI: https://doi.org/10.1016/j.csi.2012.09.004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0920548912001080.

[72] K. Pakhale, "Comprehensive overview of named entity recognition: Models, domain-specific applications and challenges", *arXiv preprint arXiv:2309.14084*, Sep. 2023. [Online]. Available: https://arxiv.org/abs/2309.14084.

[73] T. Chavan and S. Patil, "Named entity recognition (ner) for news articles", *International Journal of Artificial Intelligence Research and Development (IJAIRD)*, vol. 2, no. 1, pp. 103–112, 2024. DOI: 10.34218/IJAIRD.2.1.2024.10.

[74] G. Damnati, B. Favre, and F. Bechet, "Person name recognition and linking from overlay text in tv broadcast shows", in *Proceedings of a Conference on Multimedia Processing*, Orange Labs and Aix Marseille University LIF-CNRS, Sep. 2014.

[75] U. Zaratiana, N. Tomeh, P. Holat, and T. Charnois, *Gliner: Generalist model for named entity recognition using bidirectional transformer*, 2023. arXiv: 2311.08526. [Online]. Available: https://arxiv.org/abs/2311.08526.

[76] I. Keraghel, S. Morbieu, and M. Nadif, "Recent advances in named entity recognition: A comprehensive survey and comparative study", 2024. arXiv: 2401.10825 [cs.CL]. [Online]. Available: https://arxiv.org/abs/2401.10825.

[77] Y. Shen, "An effective sentimental analysis model based on spacy", *Highlights in Science, Engineering and Technology CSIC*, vol. 85, pp. 1065–1066, 2024.

[78] P. Gay, S. Meignier, P. Deléglise, and J.-M. Odobez, "Crf-based context modeling for person identification in broadcast videos", *Frontiers in ICT*, vol. 3, Jun. 2016. DOI: 10.3389/fict.2016.00009.

[79] P. D. Camillis, *Analysing natural language processing techniques: A comparative study of nltk, spacy, bert, and distilbert on customer query datasets*, 2022. [Online]. Available: https://arc.cct.ie/msc_da/1.

[80] V. Chakkarwar, S. Tamane, and A. Thombre, "A review on bert and its implementation in various nlp tasks", in *Proc. Int. Conf. Adv. Comput. Methods Data Anal. (ICAMIDA)*, Aurangabad, India, 2022, pp. 112–121. DOI: 10.2991/978-94-6463-136-4_12.

[81] K. Engineering, *Meet the new zero-shot ner architecture*, https://blog.knowledgator.com/meet-the-new-zero-shot-ner-architecture-30ffc2cb1ee0, Accessed: 2025-05-16, Aug. 2024.

[82] Y. Huang, K. Tang, and M. Chen, "Distilling large language models into tiny models for named entity recognition", *arXiv preprint arXiv:2402.09282*, 2024.

[83] D. Hassabis and S. Pichai, *Introducing gemini 1.5, google's next-generation ai model*, https://blog.google/technology/ai/google-gemini-next-generation-model-february-2024/, Feb. 2024.

[84] G. Team, "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context", *arXiv preprint arXiv:2403.05530*, 2024.

[85] G. Team, "Gemini: A family of highly capable multimodal models", *arXiv preprint arXiv:2312.11805*, 2024.

[86] H. Touvron *et al.*, "Llama: Open and efficient foundation language models", 2023. arXiv: 2302.13971. [Online]. Available: https://arxiv.org/abs/2302.13971.

[87] M. AI, *The llama 4 herd: The beginning of a new era of natively multimodal ai innovation*, https://ai.meta.com/blog/llama-4-multimodal-intelligence/, Apr. 2025.

[88] A. Shukla, *Llama 4: Meta's multimodal leap in ai - architecture, capabilities, and comparative analysis*, https://www.researchgate.net/publication/390581691, Apr. 2025.

[89] A. Vaswani *et al.*, "Attention is all you need", *arXiv preprint arXiv:1706.03762*, 2017.

[90] M. Valente, F. Brugnara, G. Morrone, E. Zovato, and L. Badino, "Exploring spoken language identification strategies for automatic transcription of multilingual broadcast and institutional speech", 2024. arXiv: 2406.09290 [eess.AS]. [Online]. Available: https://arxiv.org/abs/2406.09290.

[91] F. Bordes, R. Y. Pang, A. Ajay, and A. C. Li, "An introduction to vision-language modeling", *arXiv preprint arXiv:2405.17247*, 2024. [Online]. Available: https://arxiv.org/abs/2405.17247.

[92] R. Sapkota, S. Raza, M. Shoman, A. Paudel, and M. Karkee, "Multimodal large language models for image, text, and speech data augmentation: A survey", *arXiv preprint arXiv:2501.18648*, 2025. [Online]. Available: https://arxiv.org/abs/2501.18648.

[93] X. Wang, Y. Zhang, Y. Wang, and Y. Chen, “Optimizing edge ai: A comprehensive survey on data, model, and system optimization”, *arXiv preprint arXiv:2501.03265*, 2025. [Online]. Available: https://arxiv.org/abs/2501.03265.

[94] Y. Li, Z. Wang, and W. Hu, “Optimization methods, challenges, and opportunities for edge inference: A comprehensive survey”, *Electronics*, vol. 14, no. 7, p. 1345, 2025. DOI: 10.3390/electronics14071345. [Online]. Available: https://www.mdpi.com/2079-9292/14/7/1345.

[95] S. Huang, Q. Wang, and X. He, “Confidence-aware adversarial learning for self-supervised semantic matching”, *arXiv preprint arXiv:2008.10902*, 2020. [Online]. Available: https://arxiv.org/abs/2008.10902.

[96] H. Alwassel, D. Mahajan, L. Torresani, B. Ghanem, and D. Tran, “Self-supervised learning by cross-modal audio-video clustering”, in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. [Online]. Available: https://arxiv.org/abs/1911.12667.

Appendix A Dissertation Resources

All materials referenced in this dissertation are publicly accessible as follows:

Complete ANEP Pipeline

The full ANEP implementation, including comprehensive setup instructions and UI, is accessible via the following repository:
`https://github.com/AFLucas-UOM/Accurate-Name-Extraction`

The NGD

The NGD developed and utilised throughout this research can be accessed at:
`https://universe.roboflow.com/ict3909-fyp/news-graphic-dataset`

Research Survey

The participant survey conducted as part of this study is available at:
`https://aflucas.com/dissertation-form/`

Appendix B Annotation Guidelines

The annotation protocol for the NGD was meticulously designed to facilitate high-quality, frame-level labelling of graphical text overlays in diverse news video content. A corpus of 1,500 frames was manually annotated using Roboflow Annotate and exported in YOLO format to support robust downstream training and evaluation. Annotations encompass six predefined classes: *Breaking News*, *Lower Third*, *News Ticker*, *Headline*, *Digital On-Screen Graphic*, and *Other News Graphic*.

- **Bounding Box Specification:** Boxes were precisely fitted to capture the full extent of each textual graphic element, ensuring complete coverage of all characters while excluding excessive padding. For elements with complex backgrounds or integrated design features, the entire graphical unit was enclosed. This approach prioritised visual coherence and maintained the semantic integrity of each element.
- **Layering and Overlapping Elements:** When graphics of distinct classes appeared in the same region (e.g., ticker behind a lower third), each was annotated separately with overlapping bounding boxes. These were explicitly ordered in Roboflow's z-order system to reflect the visual hierarchy as presented in the original broadcast. Annotator judgement prioritised preserving both the spatial relationships and semantic clarity of overlapping elements.
- **Class Assignment Criteria:** Graphics were categorised based on a combination of:
  - Visual structure and presentation style
  - Screen position and spatial relationship to other elements
  - Semantic role within the broadcast narrative
  - Temporal persistence (particularly for breaking news banners)

  Channel branding elements (corner logos, station identifiers) were consistently classified as *Digital On-Screen Graphic*. Elements that defied conventional categorisation, particularly in social media samples with non-standard layouts, were assigned to *Other News Graphic*.
- **Domain-Specific Job Structuring:** Annotations were organised into three distinct labelling jobs to reflect the source domain and accommodate medium-specific presentation styles:
  - **ForeignNews:** 606 frames sampled from international broadcast networks, including major global news services and regional broadcasters from diverse geographical regions.

  - **TikTokNews:** 594 frames extracted from short-form social media news content, capturing both established news outlets' TikTok presence and independent news creators.
  - **LocalNews:** 300 frames from Maltese broadcast news, including primary national channels and specialised local news programmes.

- **TikTok-Specific Labelling Protocol:** For TikTok-derived samples, additional considerations were implemented:
  - All overlays with clear journalistic intent (identifying sources, subjects, topics) were prioritised as *Lower Third* or *Headline* depending on layout and function
  - Creator-added text elements serving as commentary were classified by their positioning and function rather than aesthetic presentation
  - Platform-native features (e.g., TikTok captions) were distinguished from creator-added elements
  - Non-standard text overlays and purely decorative elements were labelled as *Other News Graphic*

- **Exclusion and Edge Cases:** The following elements were systematically excluded from annotation:
  - Minimal channel watermarks lacking substantive informational content
  - Background imagery not directly integrated with text elements
  - Incidental on-screen text (e.g., visible street signs in footage)

- **Annotation Environment:** All labelling was conducted using Roboflow Annotations, with standardised display settings and consistent zoom levels to ensure uniform treatment of elements across the corpus. Screen resolution and colour calibration were standardised across annotation sessions.

This annotation methodology prioritised consistency whilst acknowledging the visual diversity of contemporary news graphics across broadcast television, streaming platforms, and social media channels. The resulting dataset provides a comprehensive foundation for developing robust text detection models capable of generalising across the evolving landscape of news media presentation formats.

Appendix C DFD & Flowcharts

This appendix begins with a simplified flow diagram of the ANEP, offering a clear overview of the core processing stages from video input to name extraction and summarisation. This is followed by the level 1 DFD, which outlines the internal components and external interfaces. Subsequent process flowcharts capture detailed user-facing workflows, frontend–backend interactions, system-level pipelines, and the operation of both traditional and GenAI extraction modules.

C.1 Overview of ANEP

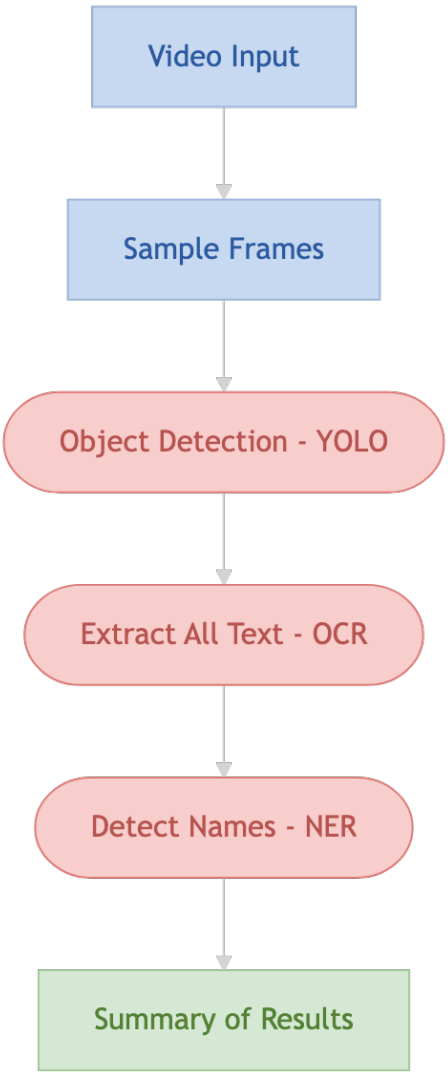


Figure C.1 Simplified workflow of ANEP

C.2 Level 1 DFD of the ANEP UI System

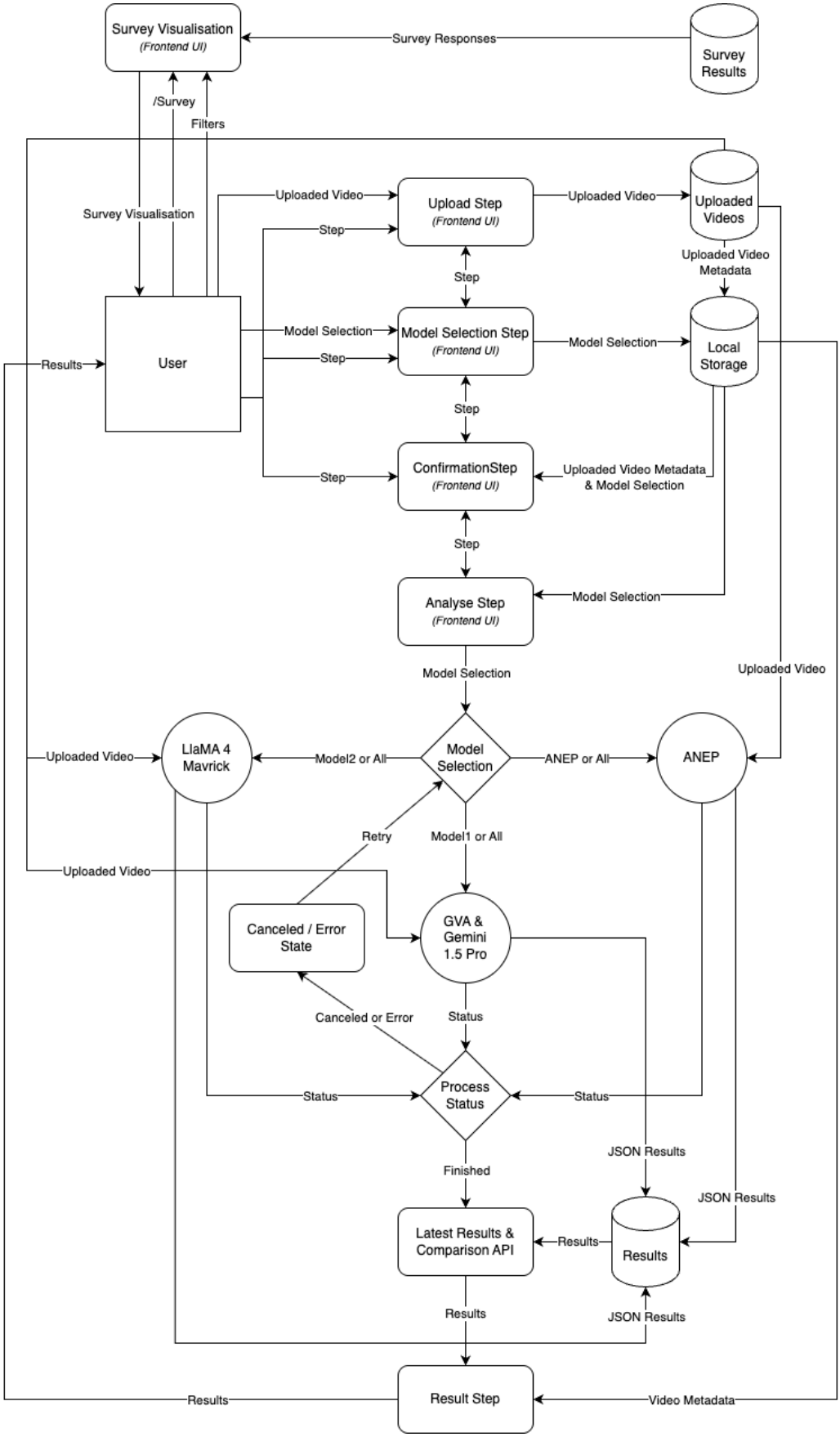


Figure C.2 Level 1 DFD of ANEP UI

C.3 ANEP UI Flowchart

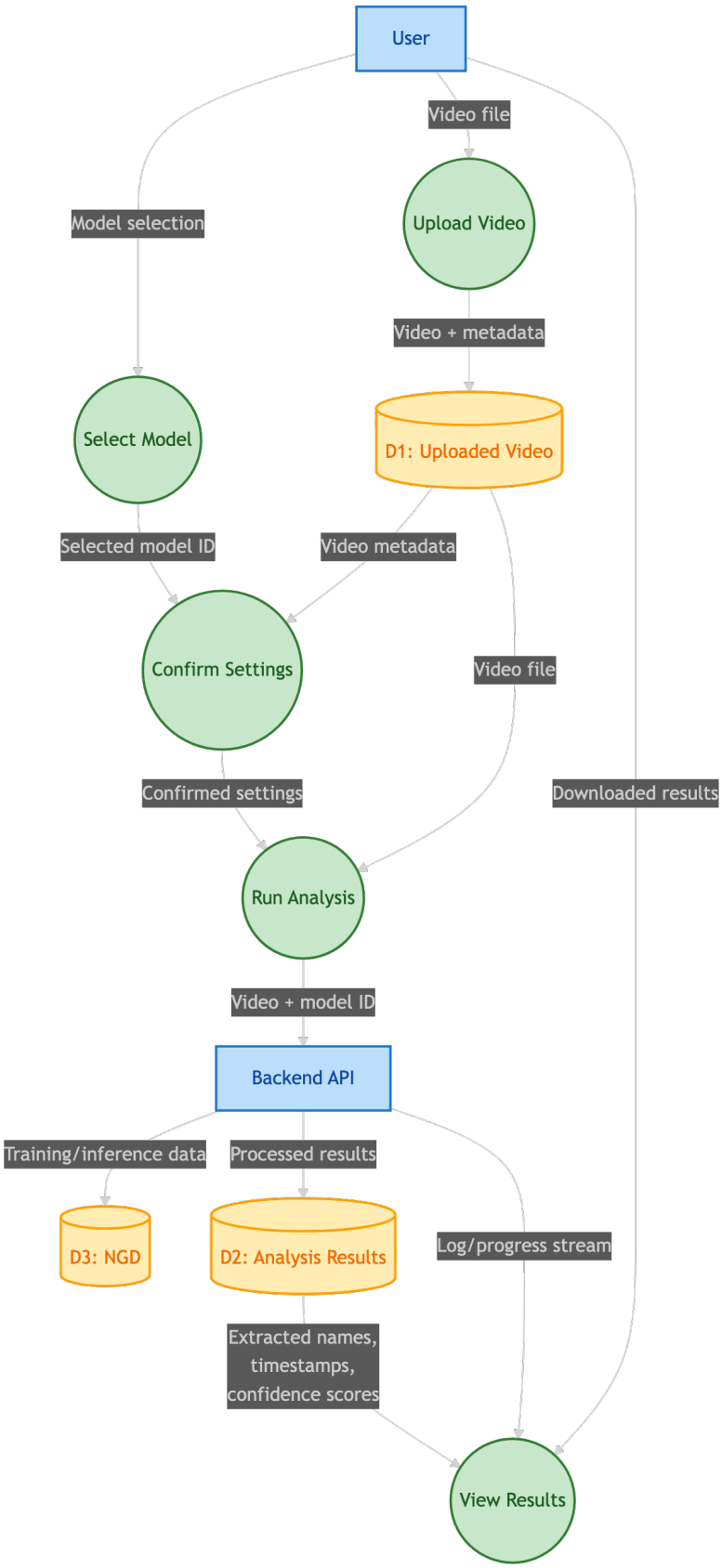


Figure C.3 ANEP UI Workflow

C.4 ANEP Flowchart

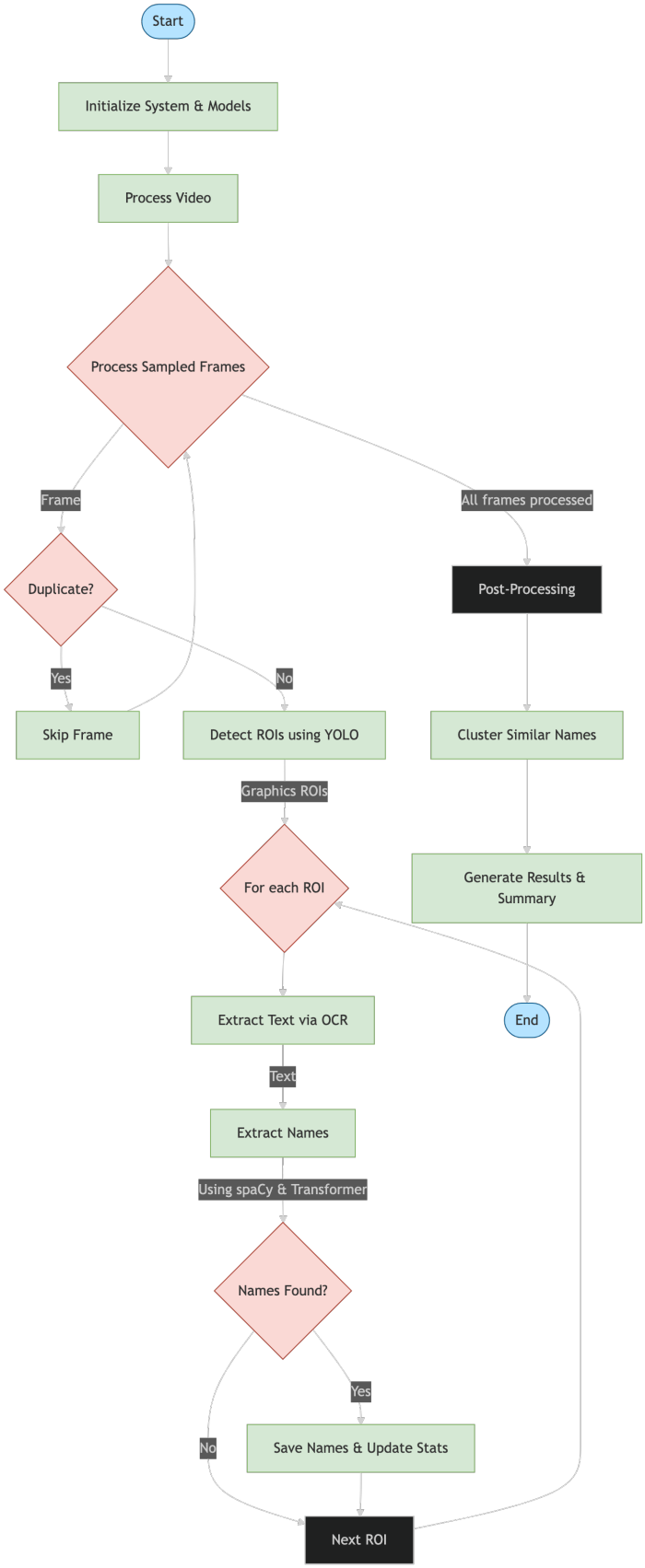


Figure C.4 ANEP System Architecture

C.5 GVA Flowchart

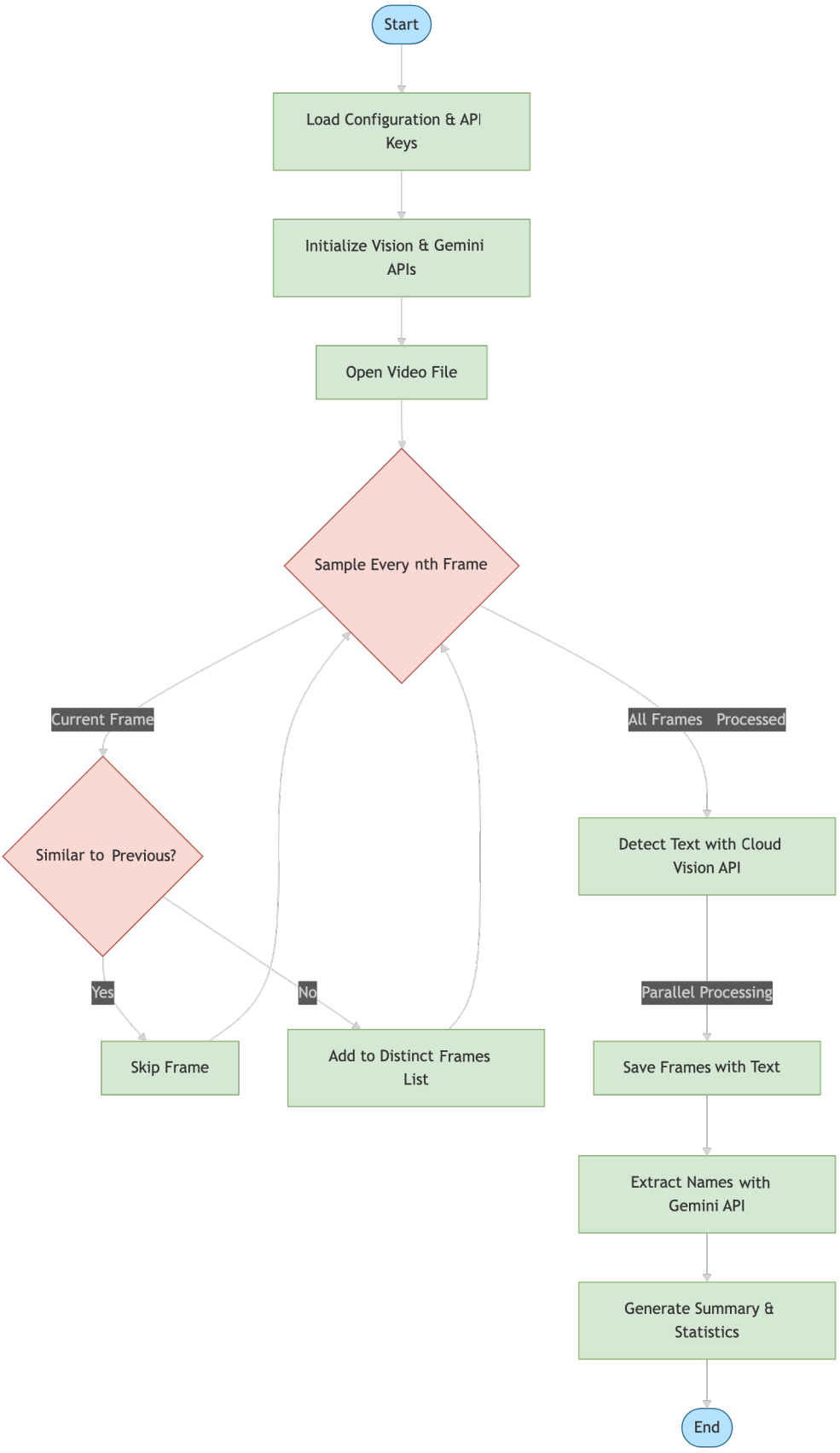


Figure C.5 GVA Workflow

C.6 LLaMA Flowchart

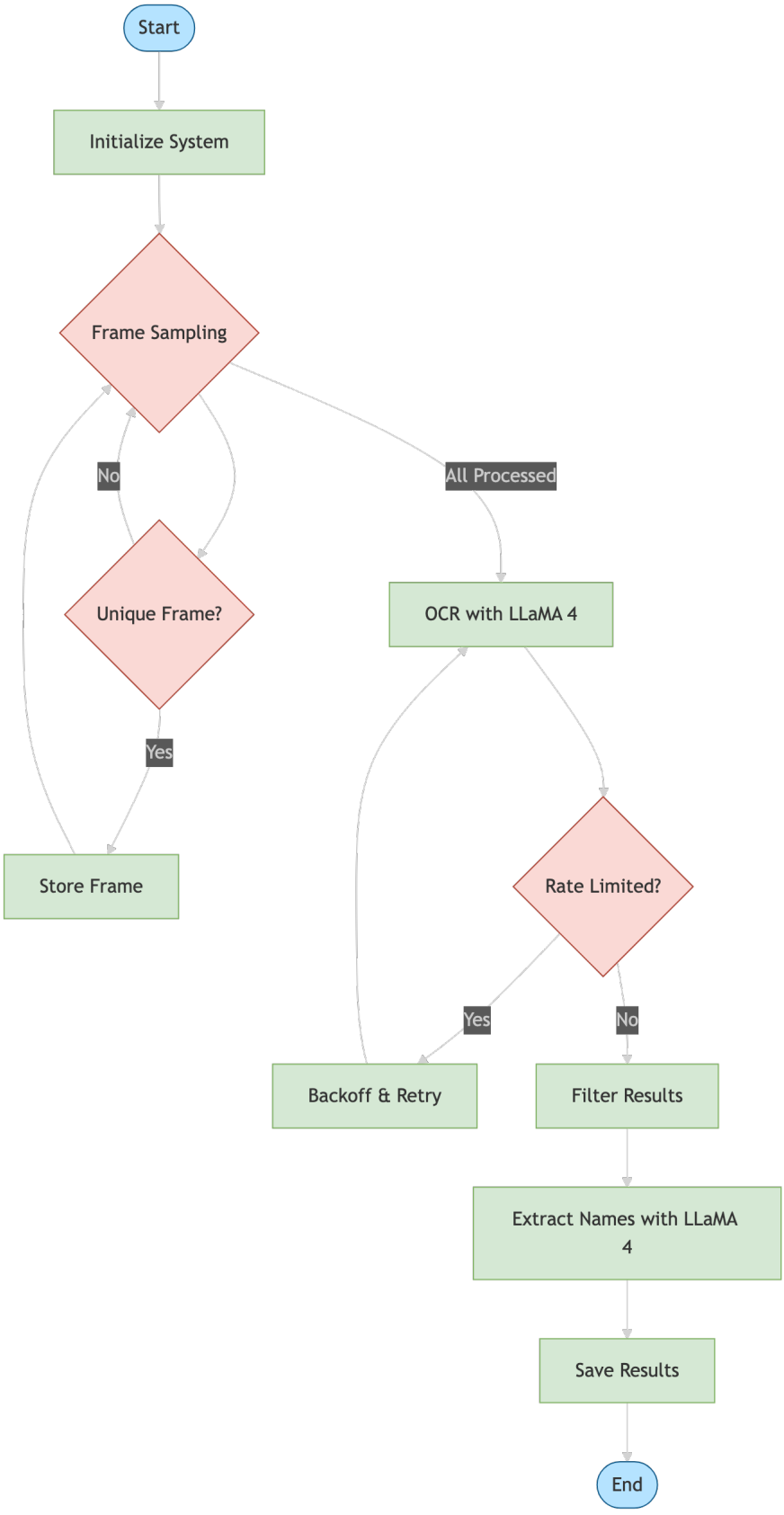


Figure C.6 LLaMA Workflow

Appendix D Prompt Templates

This appendix provides detailed documentation of the prompt templates used in the multimodal analysis pipeline for name extraction and OCR.

D.1 Gemini Prompt for Name Extraction

The following prompt is used with the Gemini 1.5 Pro LLM to extract real-world personal names from aggregated OCR snippets:

```
I need to extract proper names (people's names) from the text data from
a video.
The text data is from multiple frames and is as follows:
{JSON array of timestamped text snippets}
Analyse the text and extract ONLY real people's names. Skip:
- Brand names
- Channel names
- Show names
- App names
- Non-name text
Return the result as a JSON object with this exact format:
{
  "names": [
    {
      "name": "Full Name",
      "first_appearance": "timestamp",
      "last_appearance": "timestamp",
      "count": number_of_appearances
    }
  ]
}
Only include valid people's names. Return an empty array if no real people's
names are found. Be precise and return ONLY the JSON object, nothing else.
```

D.2 LLaMA Prompts for OCR

A dual-prompt strategy is employed when processing video frames through the LLaMA model via the OpenRouter API. Each frame is base64-encoded and processed with a primary prompt, with a fallback prompt deployed when initial text extraction yields insufficient results.

D.2.1 Primary Prompt

The primary prompt is designed to be concise yet effective for standard text extraction scenarios:

```
Extract and return only visible text from this image. Respond with plain
text only.
```

D.2.2 Fallback Prompt

When the primary prompt yields inadequate results, the system automatically implements this more detailed instruction set:

```
This image may contain hard-to-read text. Please analyse carefully and
extract ALL visible text, even if it's partially obscured or in unusual
positions. Be thorough and focus on any text that might be present.
Respond with ONLY the extracted text, nothing else.
```

Appendix E Additional Evaluation Results

This appendix presents additional dataset statistics, qualitative visualisations and detailed evaluation metrics for the final OD model.

E.1 NGD Dataset: Statistical Breakdown

The figures below include co-occurrence patterns, class distribution, spatial density heat maps, object frequency per image, and bounding box centre distributions. These breakdowns help to verify data diversity, spatial bias, and class overlap—critical considerations for training robust deep learning models.

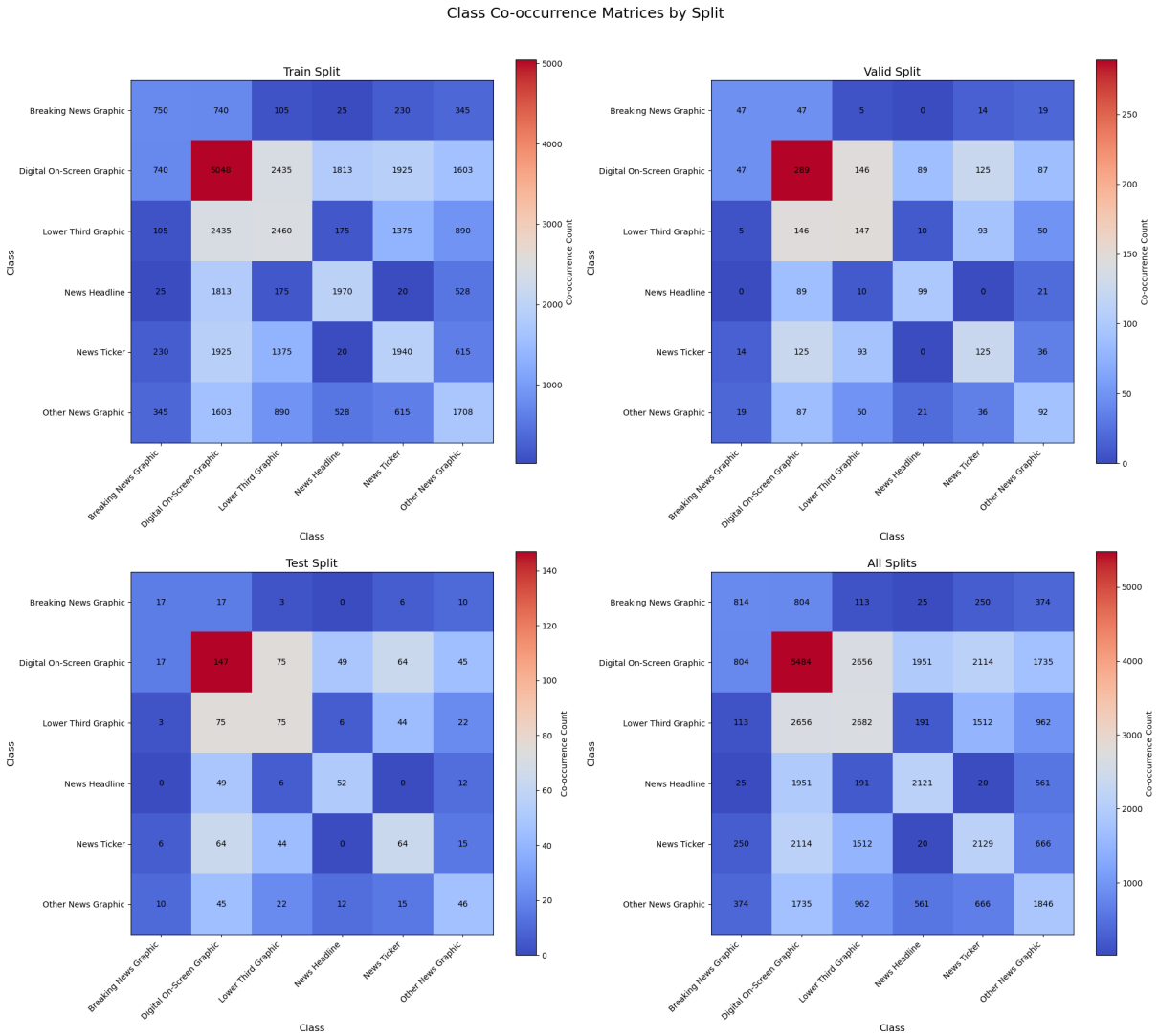


Figure E.1 Class co-occurrence matrices by split, showing how frequently each class appears alongside others.

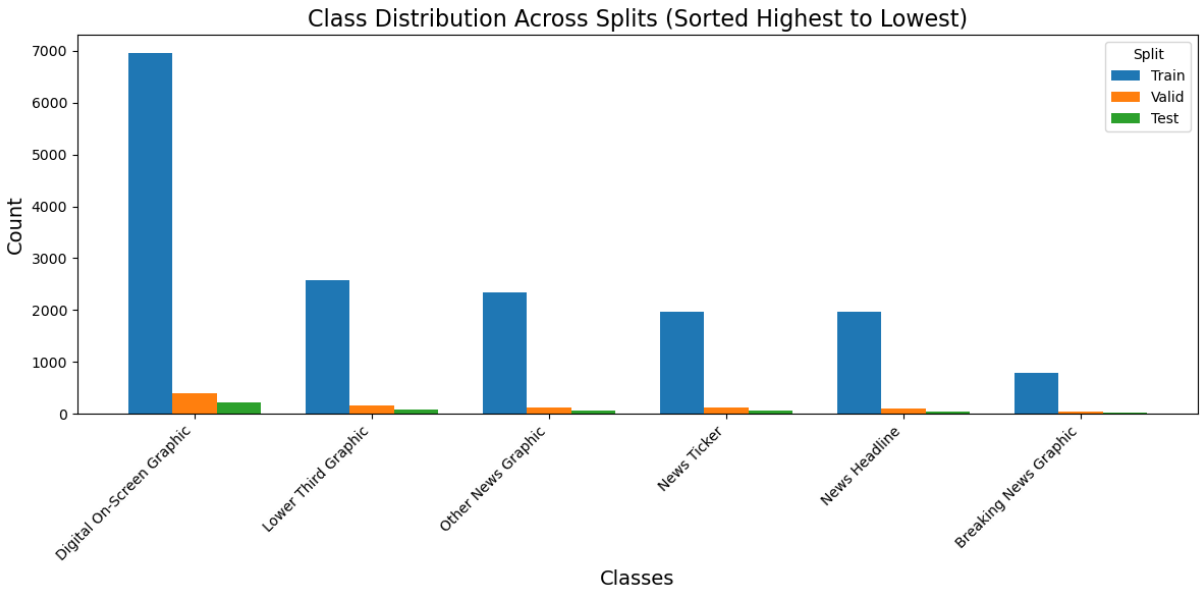


Figure E.2 Class distribution across all splits, sorted by total frequency.

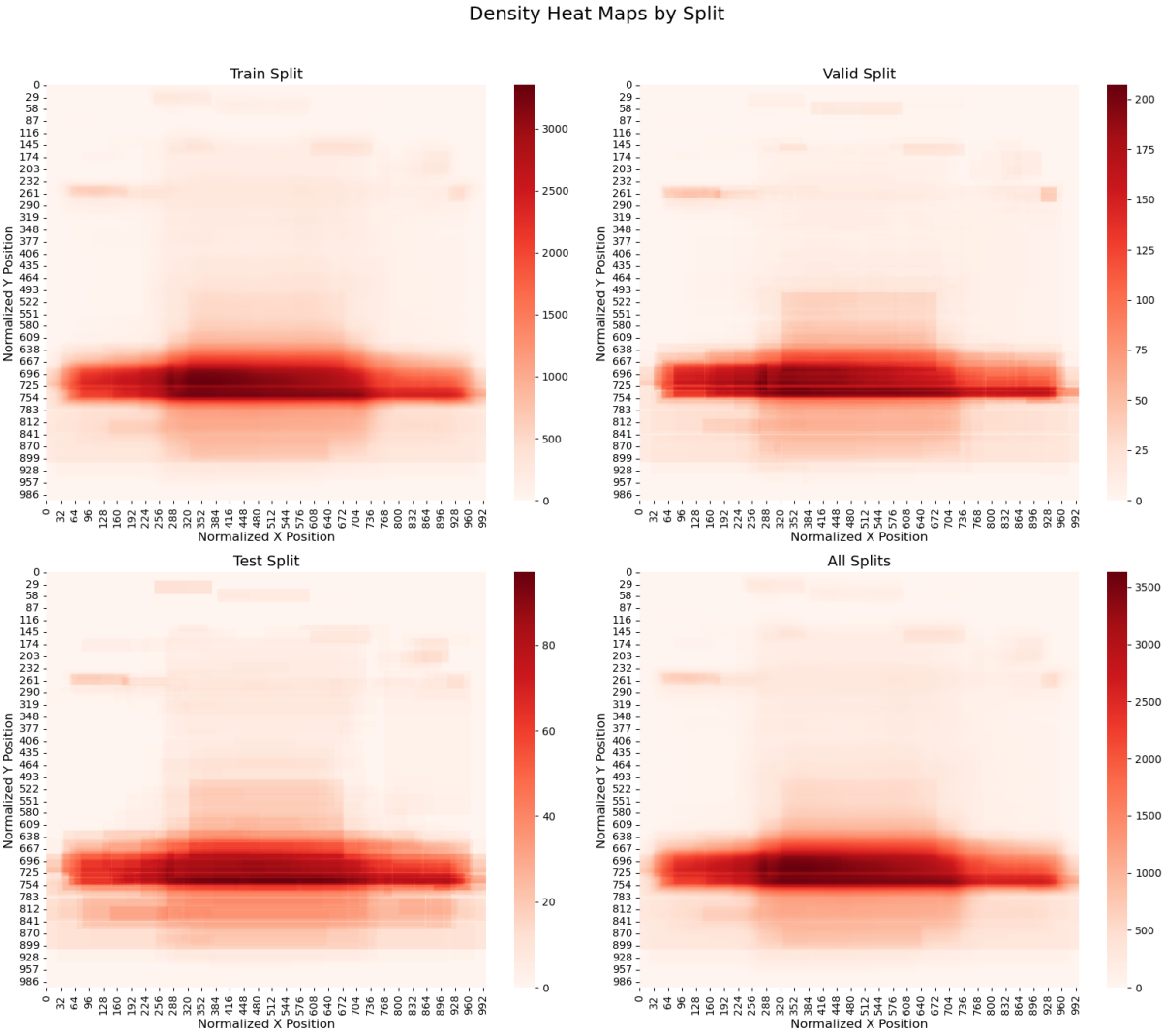


Figure E.3 Heat maps showing spatial density of object annotations across splits. Strong horizontal bands reflect common placement of on-screen graphics.

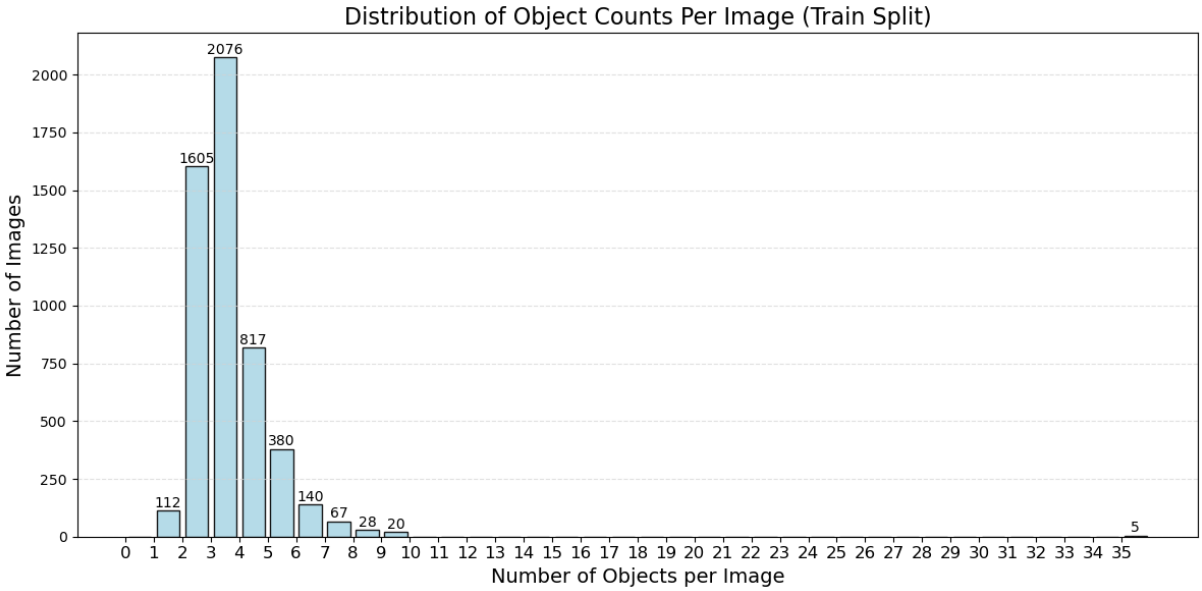


Figure E.4 Histogram showing the distribution of object counts per image in the training split. Most images contain between 3 and 5 graphic elements.

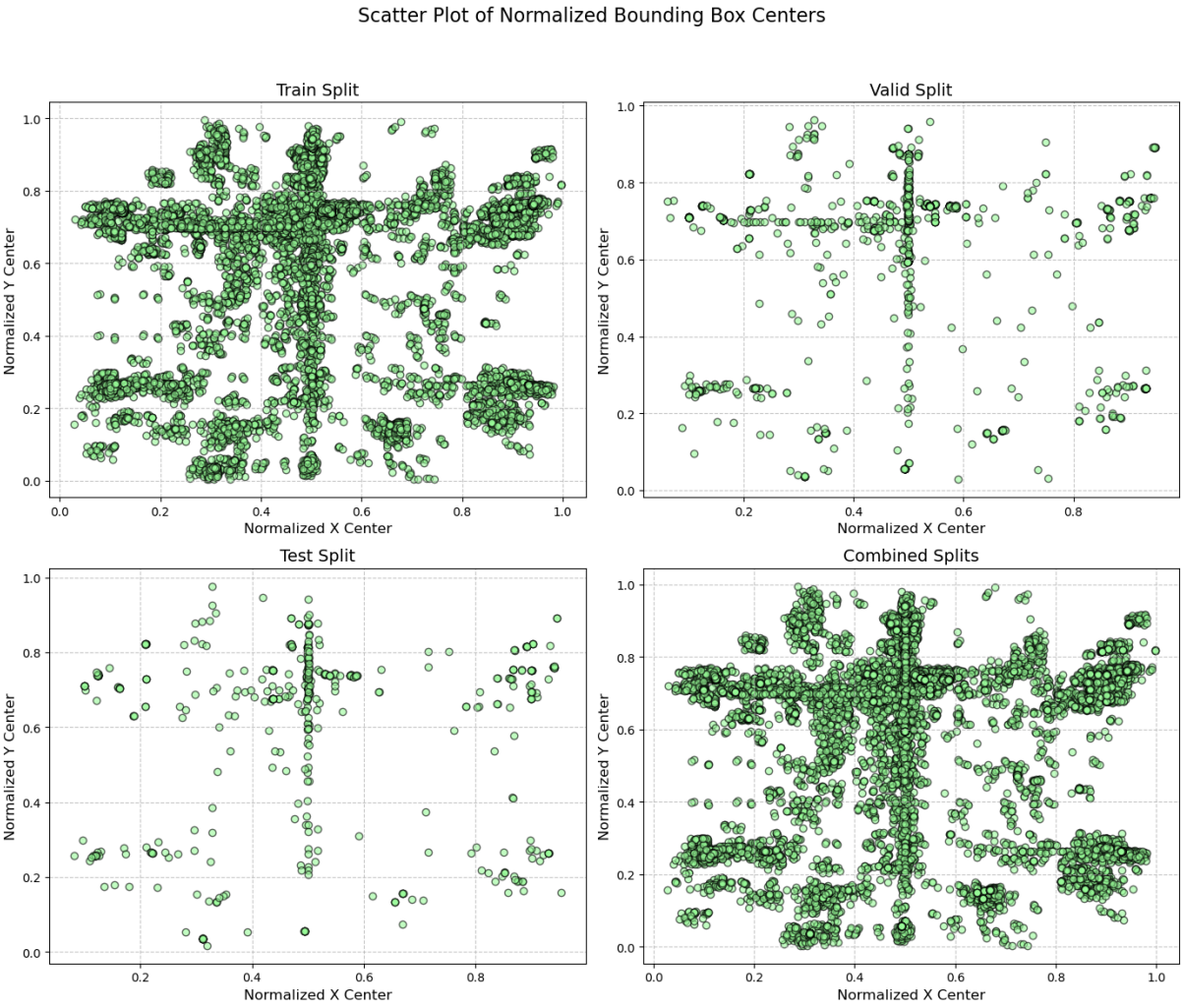


Figure E.5 Scatter plots of normalised bounding box centres across dataset splits. Clustered horizontal and vertical alignments reflect standardised broadcast layouts.

E.2 YOLOv12: Further Visualisation Examples

This section presents qualitative examples of the `NGD-YOLOv12_v5` model applied to unseen test frames. Image pairs contrast original frames against model predictions, displaying detected graphics, class labels, and confidence scores.

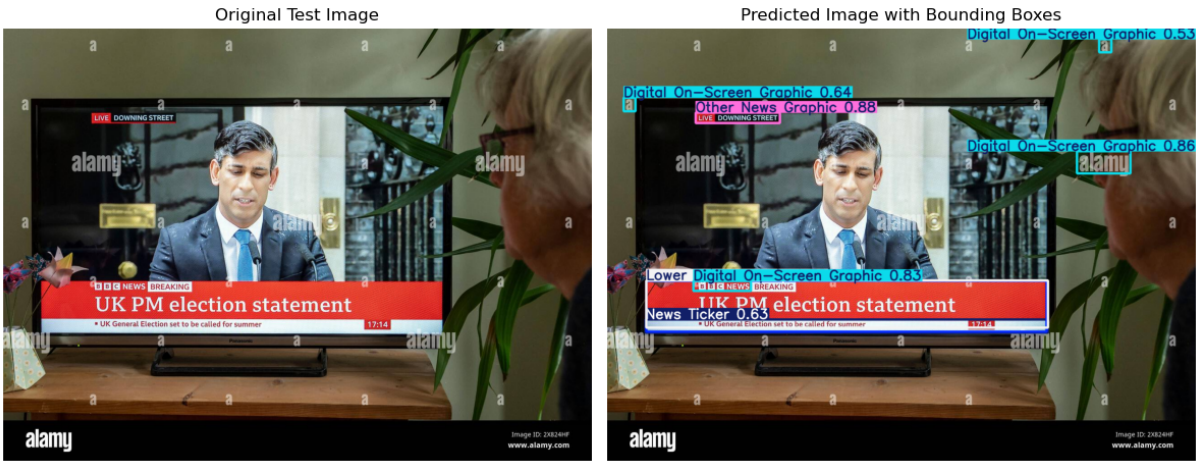


Figure E.6 Example 1 — BBC broadcast shown on a TV screen. The model accurately detects overlapping graphics despite the indirect angle, though it misclassifies the `Breaking News` banner as a `Lower Third Graphic`.

Figure E.7 Example 2 — CNN studio broadcast. Most graphics are correctly detected, but the `Breaking News` label is again misclassified as a `Lower Third Graphic`, likely due to class imbalance in the NGD.

E.3 YOLOv12: Training and Evaluation Metrics

This section presents extended performance visualisations for the `NGD-YOLOv12_v5` model. The figures include confusion matrices, metric curves, and performance breakdowns across training epochs and confidence thresholds. These complement the main evaluation by offering a more granular view of classification quality, stability, and threshold sensitivity.

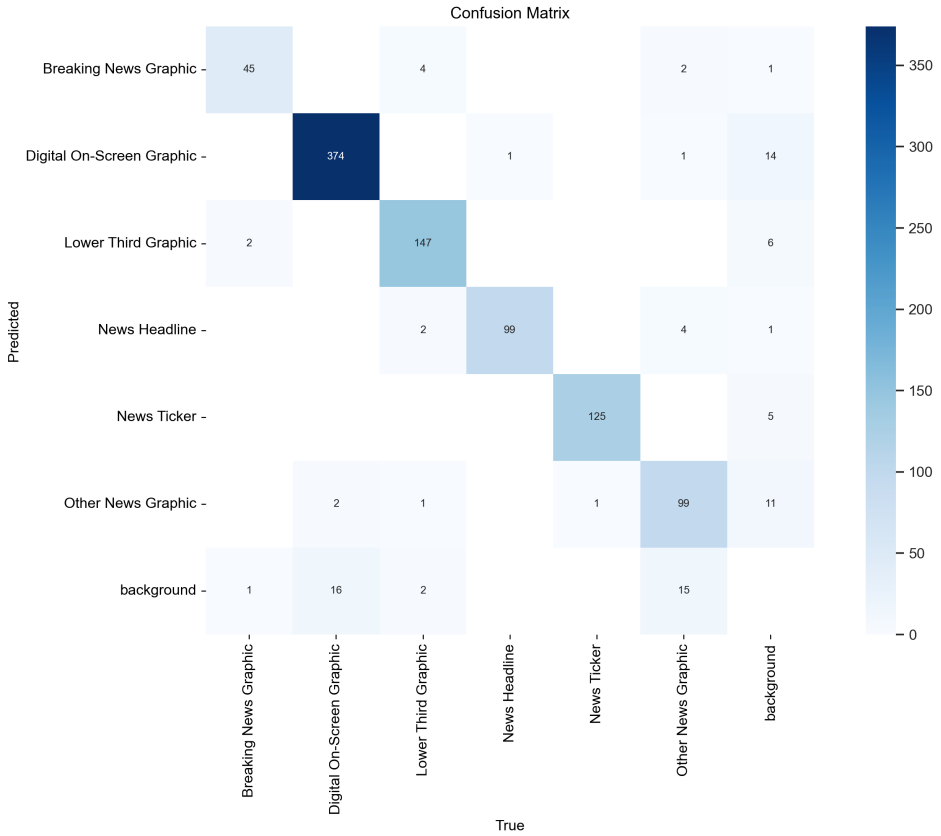


Figure E.8 Raw confusion matrix showing class-wise prediction counts for the `NGD-YOLOv12_v5` model.

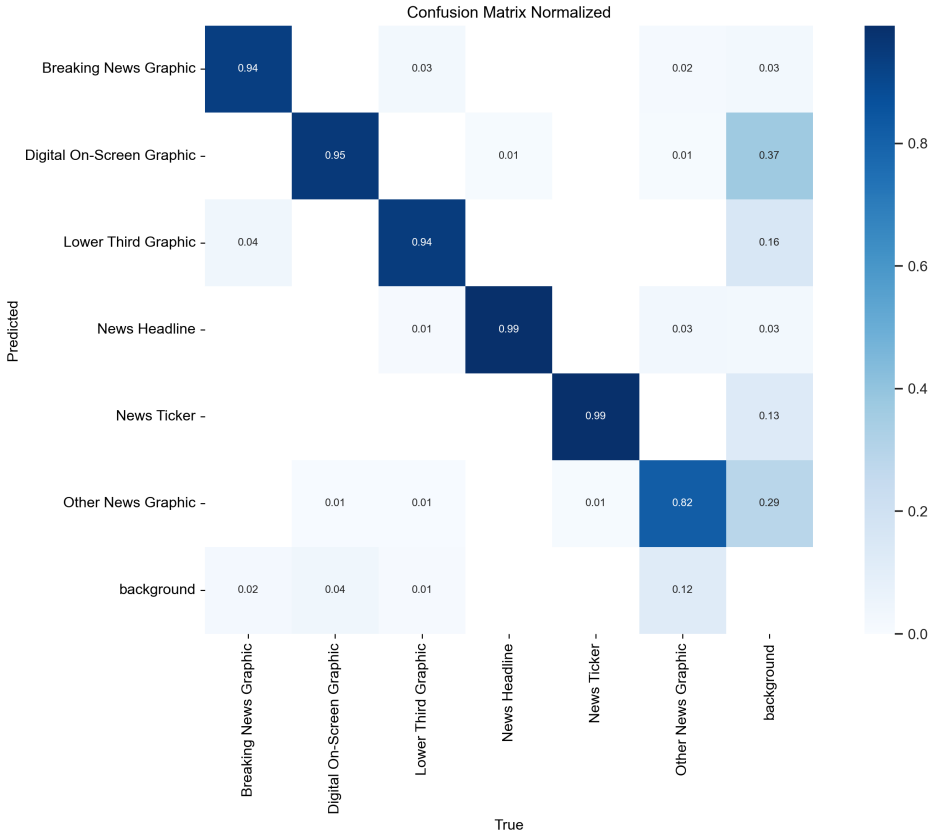


Figure E.9 Normalised confusion matrix, highlighting relative misclassifications and inter-class confusion.

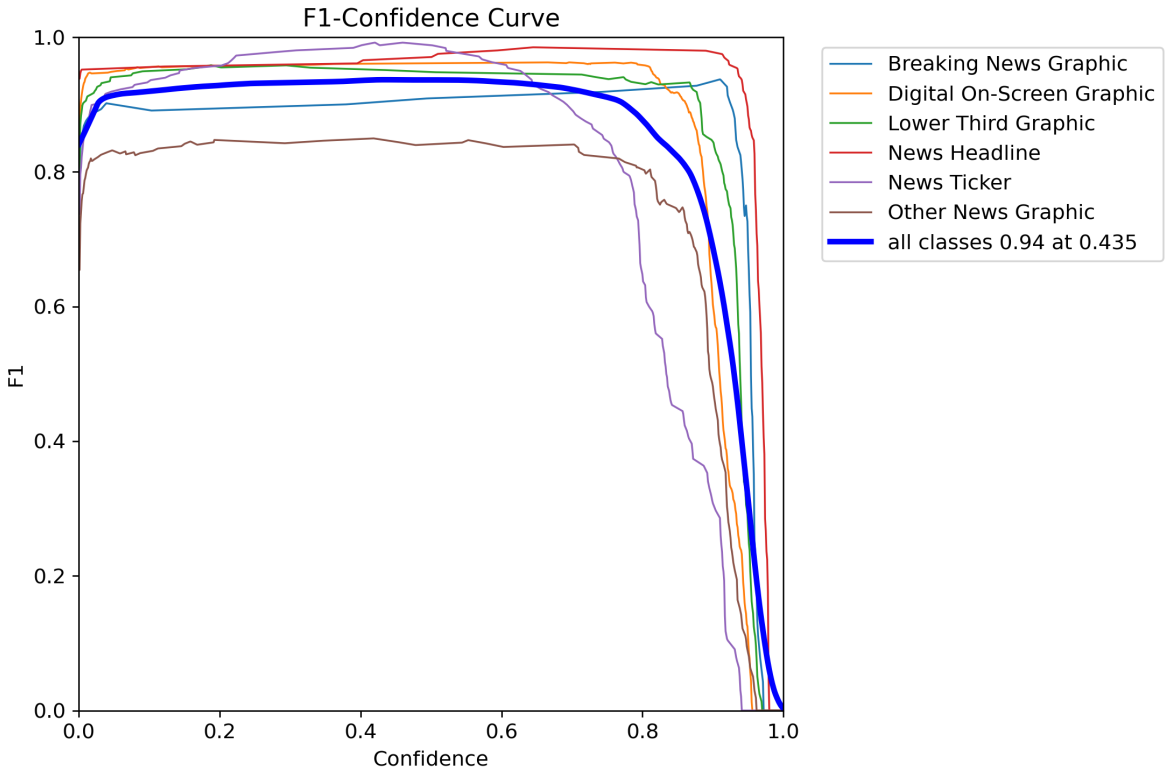


Figure E.10 F1 score vs confidence threshold for each class, showing optimal threshold (0.435) achieving 94% global F1.

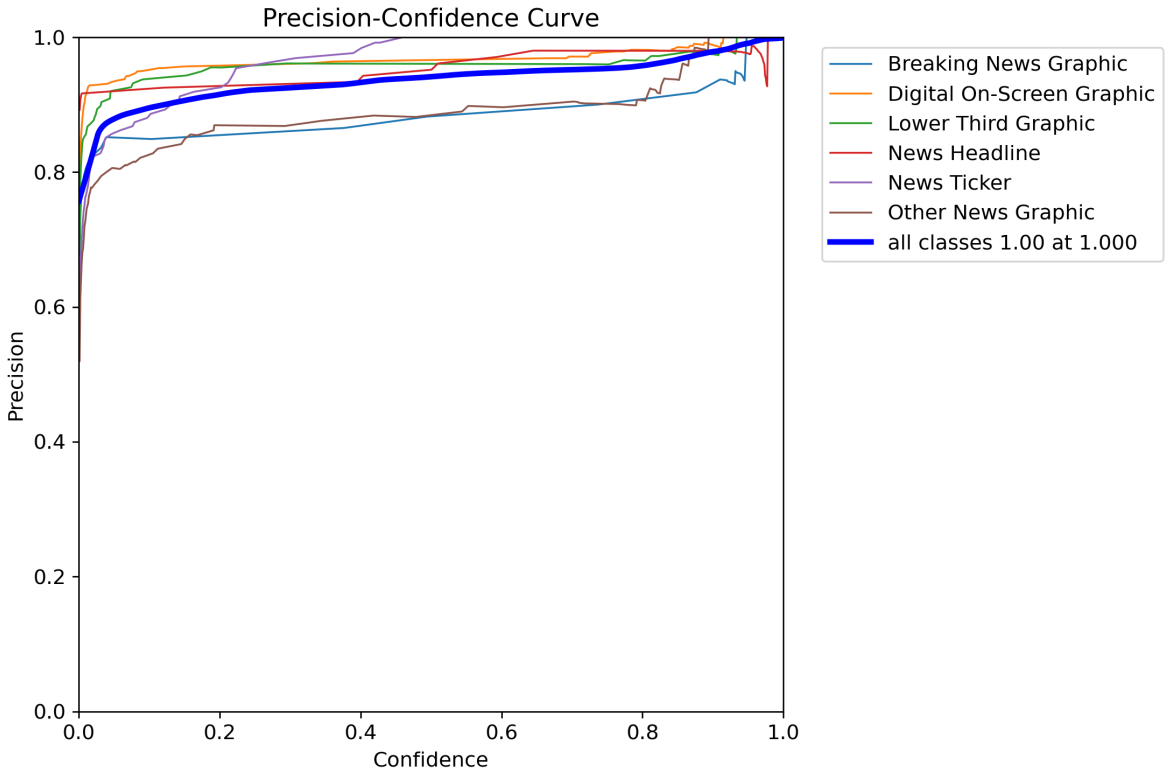


Figure E.11 Precision vs confidence curve across all classes, reflecting class-specific reliability at varying thresholds.

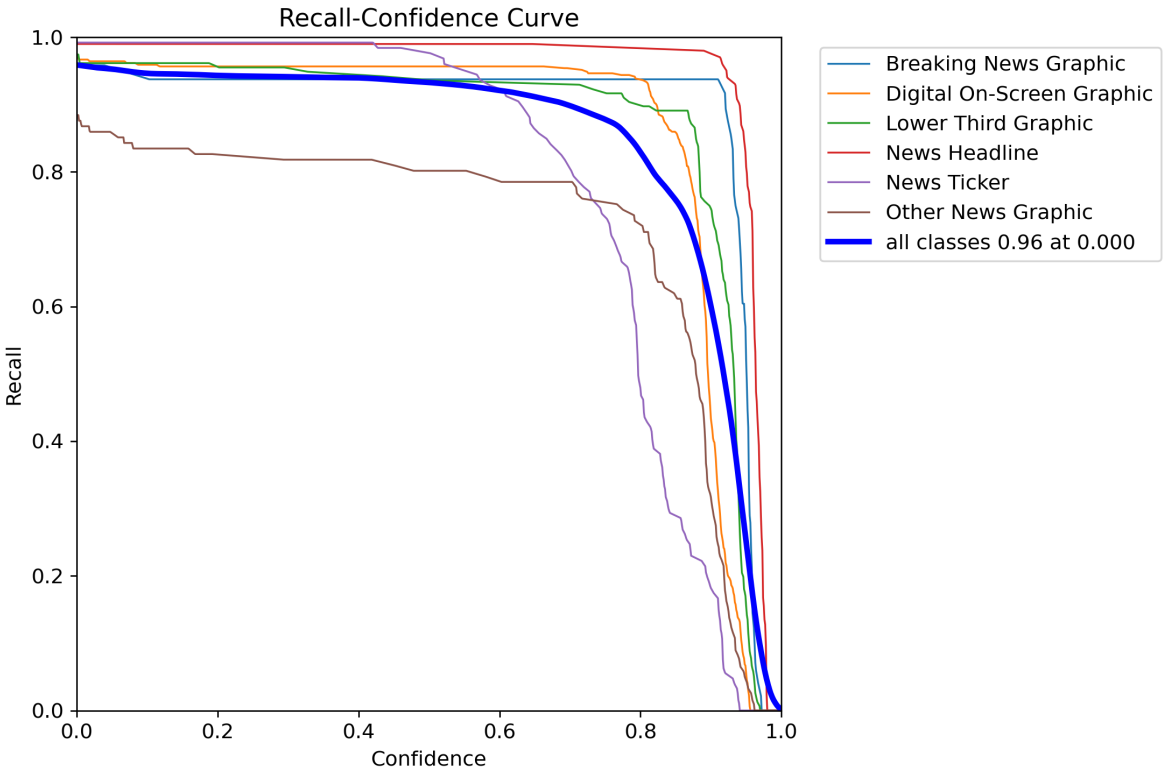


Figure E.12 Recall vs confidence curve showing detection sensitivity trends by class.

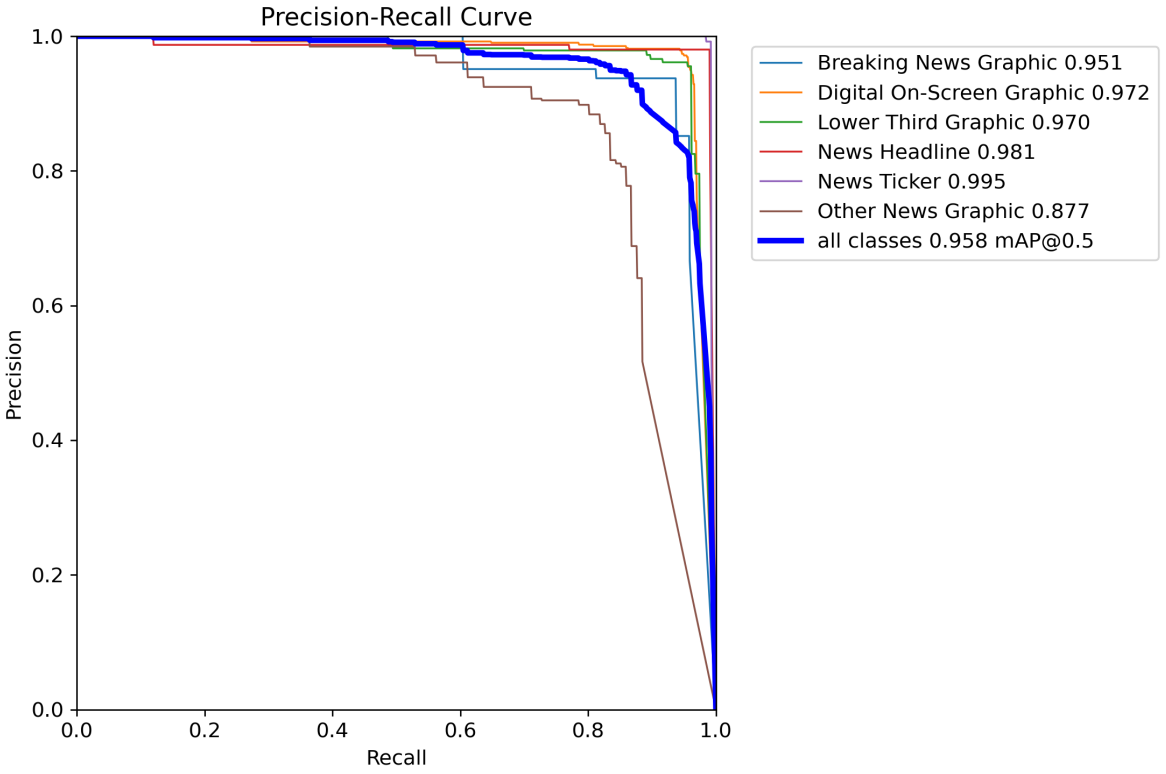


Figure E.13 Precision–recall curves per class, with mAP@0.5 reaching 95.8% overall.

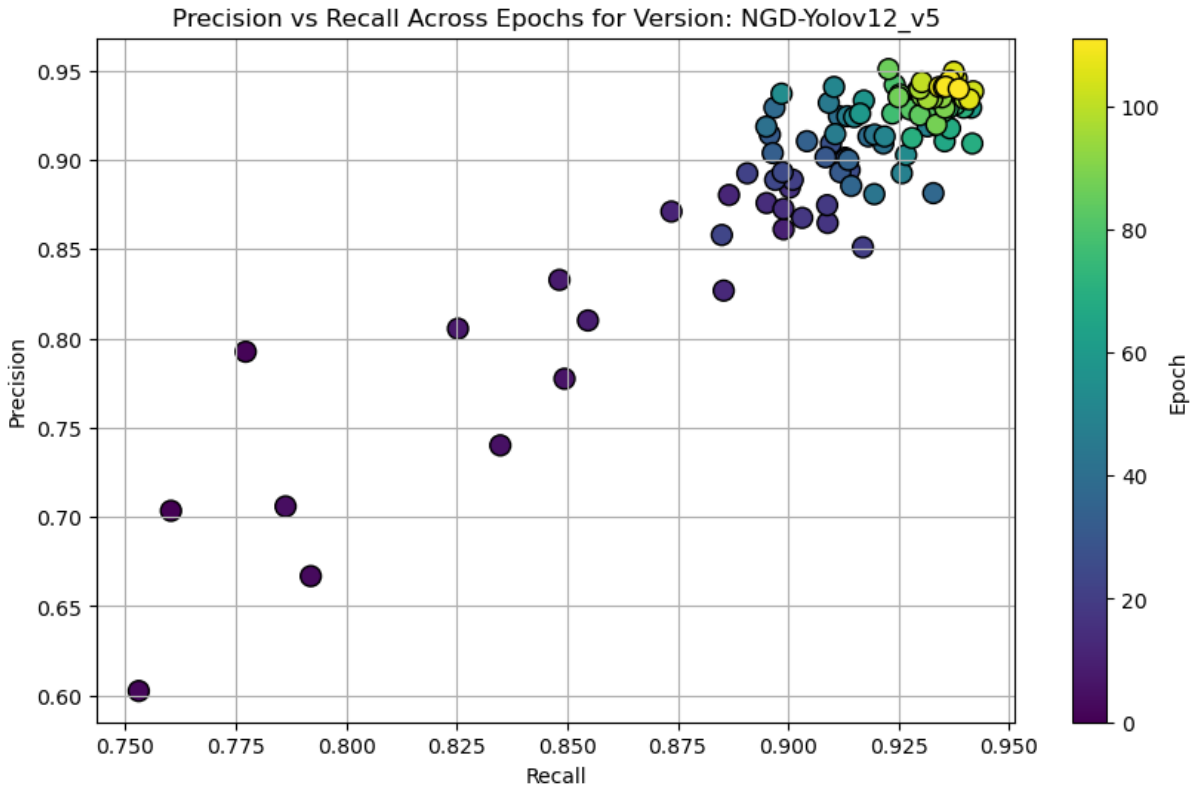


Figure E.14 Scatter plot of precision vs recall across training epochs, colour-coded by epoch number.

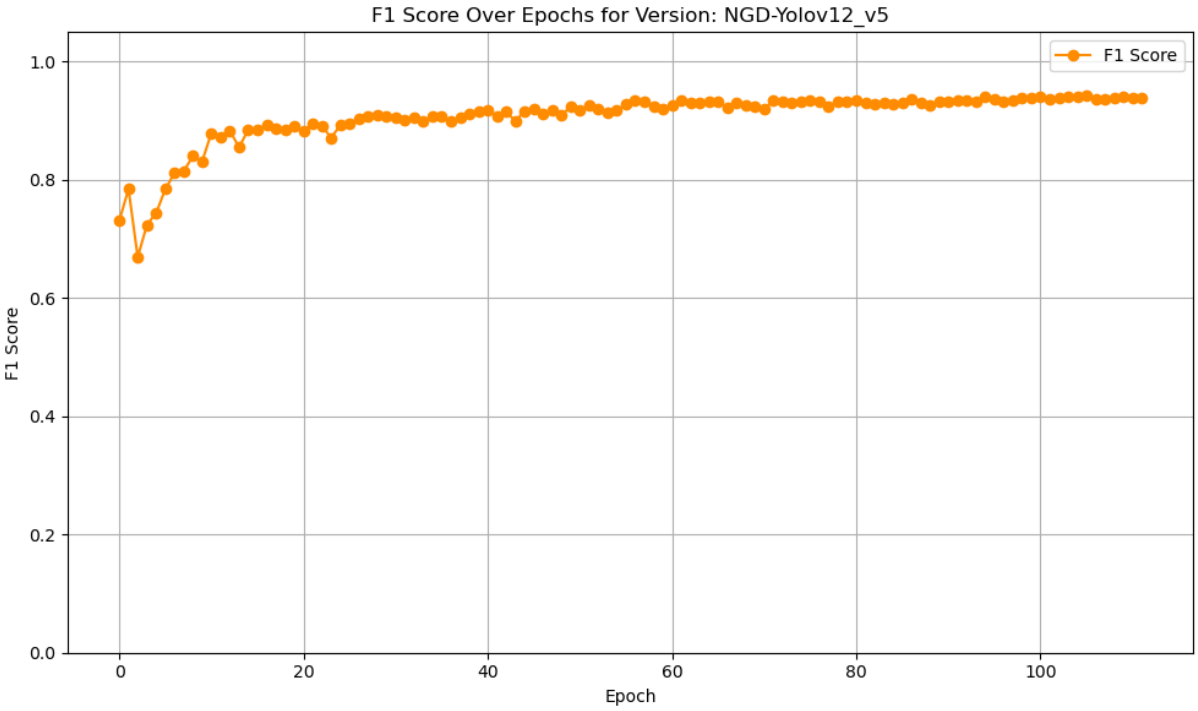


Figure E.15 F1 score progression over training epochs, showing convergence near 93%.

Appendix F NGD Breakdown

This appendix provides a structured overview of the video sources used to construct the NGD, grouped by platform domain. The dataset encompasses short-form social media, international broadcasters, and local (Maltese) news broadcasts.

F.1 Overview of NGD

Domain	Total Videos
TikTok News	120
Foreign News	120
Local News	60
Total (All Sources)	**300**

Table F.1 Summary of video counts grouped by domain

Domain	Annotated Frames
TikTok News	594
Foreign News	606
Local News	300
Total (All Sources)	**1,500**

Table F.2 Summary of annotated image frames per domain in the NGD

F.2 Video Downloads by Source

TikTok Sources	Number of Videos
SideStreet	15
MaltaDaily	15
LovinMalta	15
DailyMail	15
SkyNews (TikTok)	15
TVMNews (TikTok)	15
BBCNews (TikTok)	15
FoxNews (TikTok)	15
Total (TikTok)	**120**

Foreign News Sources	Number of Videos
C-SPAN	15
CNN	15
BBC	15
France24	15
ABCNews	15
CBS	15
FoxNews (Broadcast)	15
SkyNews (Broadcast)	15
Total (Foreign)	**120**

Local News Sources	Number of Videos
TVM News	20
One News	20
Net News	20
Total (Local)	**60**

Table F.3 Number of videos downloaded per source, grouped by content domain

F.3 Annotated Frames by Source

TikTok Sources	Annotated Frames
SideStreet	75
MaltaDaily	72
LovinMalta	74
DailyMail	74
SkyNews (TikTok)	74
TVMNews (TikTok)	75
BBCNews (TikTok)	75
FoxNews (TikTok)	75
Total (TikTok)	**594**

Foreign News Sources	Annotated Frames
C-SPAN	75
CNN	75
BBC	78
France24	75
ABCNews	78
CBS	75
FoxNews (Broadcast)	75
SkyNews (Broadcast)	75
Total (Foreign)	**606**

Local News Sources	Annotated Frames
TVM News	102
One News	97
Net News	101
Total (Local)	**300**

Table F.4 Number of annotated frames per source, grouped by content domain

Appendix G Frame Extraction & ANEP UI

G.1 Frame Extraction GUI

The frame extraction GUI was developed to allow users to visually inspect and select candidate frames from input news videos. The application automatically detects a set of key frames and presents them to the user for confirmation or replacement. This allows for manual curation to ensure that the selected frames contain high-quality and relevant graphic regions.

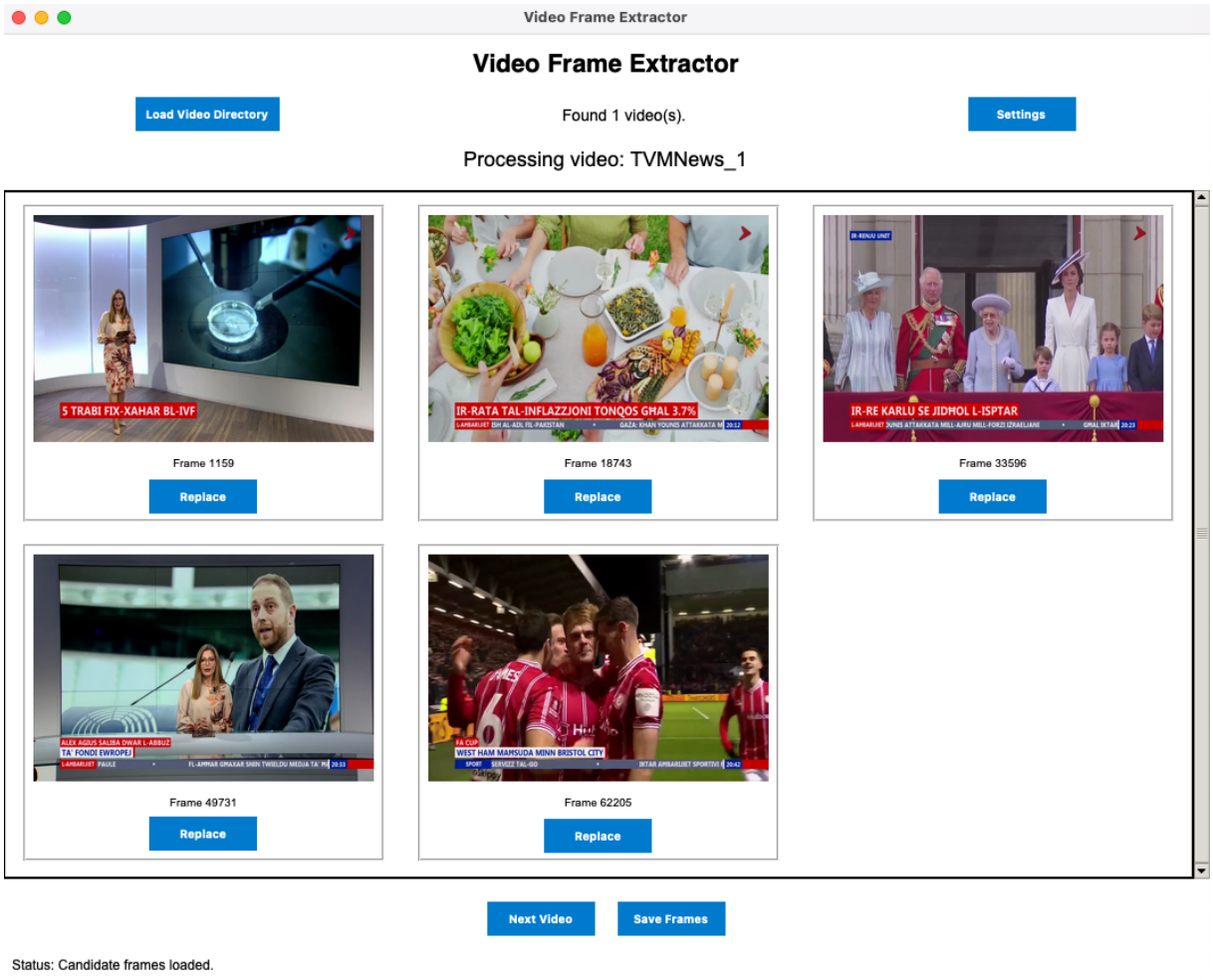


Figure G.1 Frame selection interface showing extracted candidate frames from a TVM news video - Each frame is accompanied by a frame number and the option to replace it, allowing manual refinement prior to analysis

G.2 ANEP's Web UI

The web interface for the ANEP guides users through the full pipeline in a user-friendly, step-based workflow. Built as a modern web application, the UI enables seamless upload, model configuration, analysis, and result exploration.

G.2.1 Upload Step

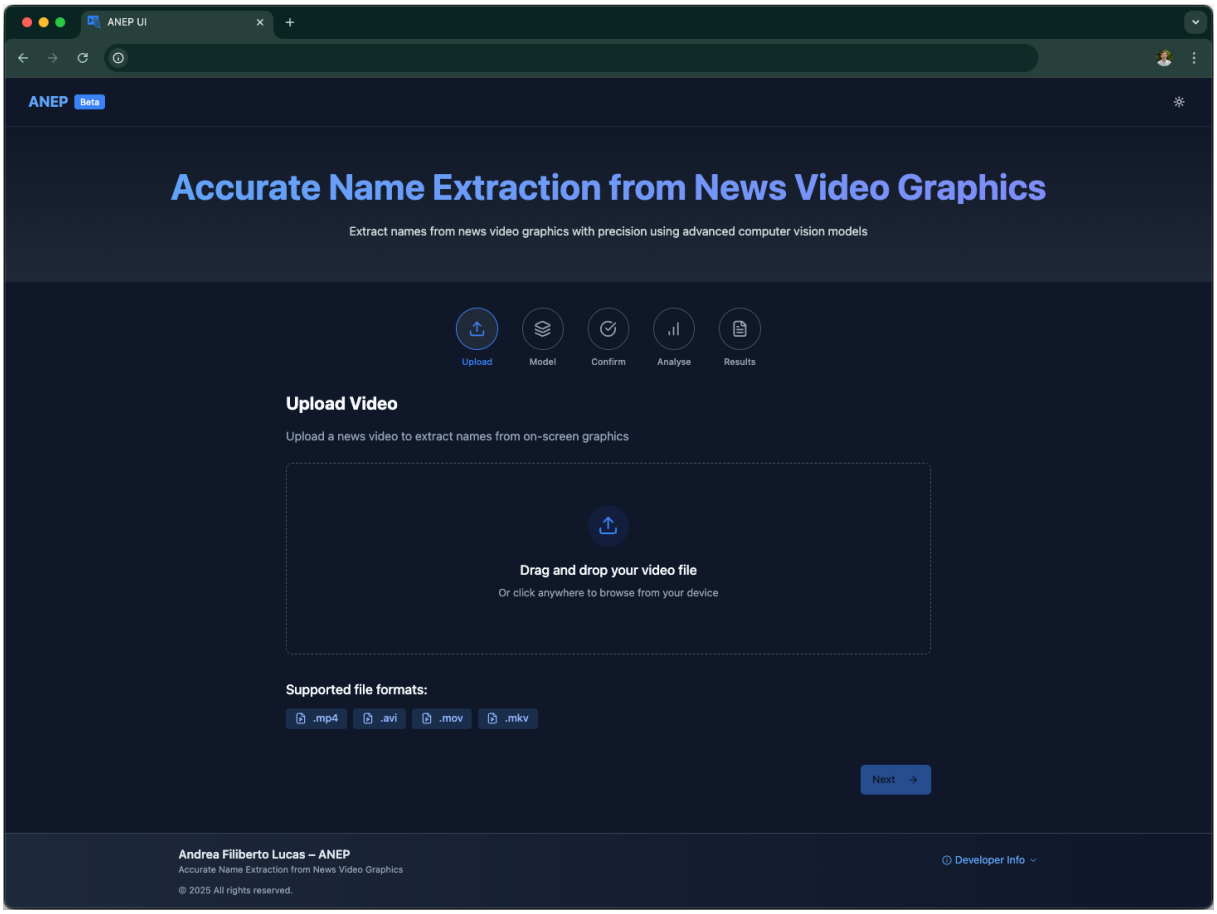


Figure G.2 Upload interface in dark mode - Users can drag and drop a video file to initiate the pipeline

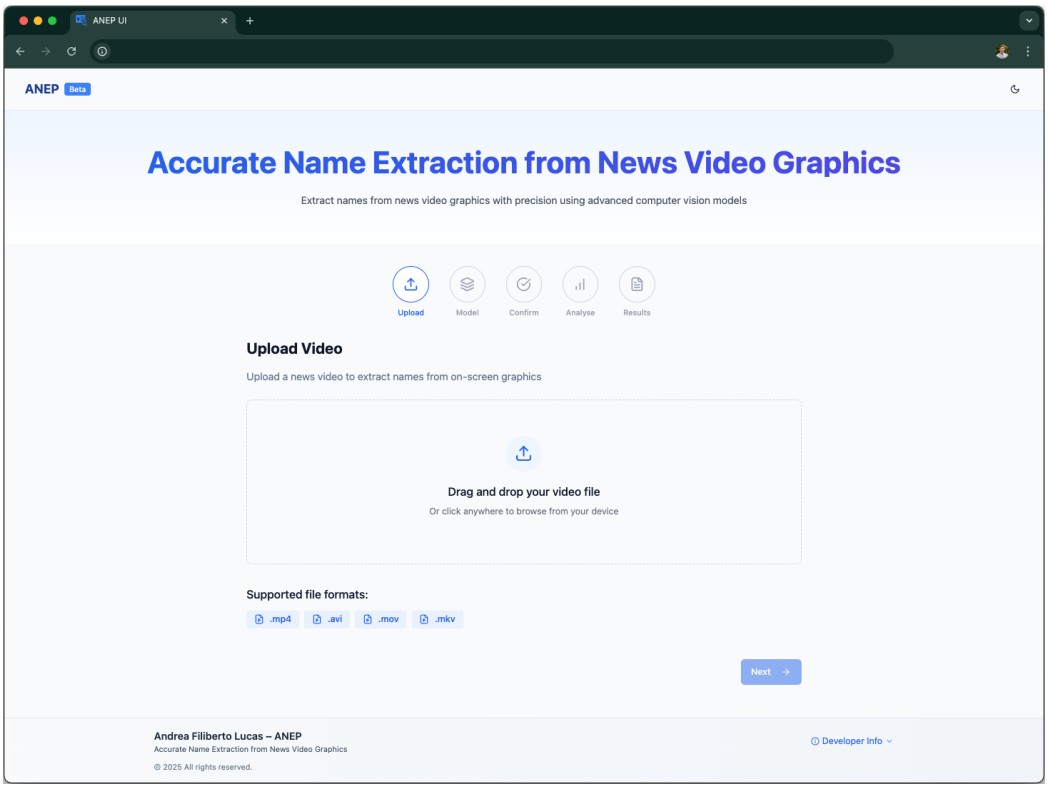


Figure G.3 Upload interface in light mode - Same functionality presented with an alternative visual theme

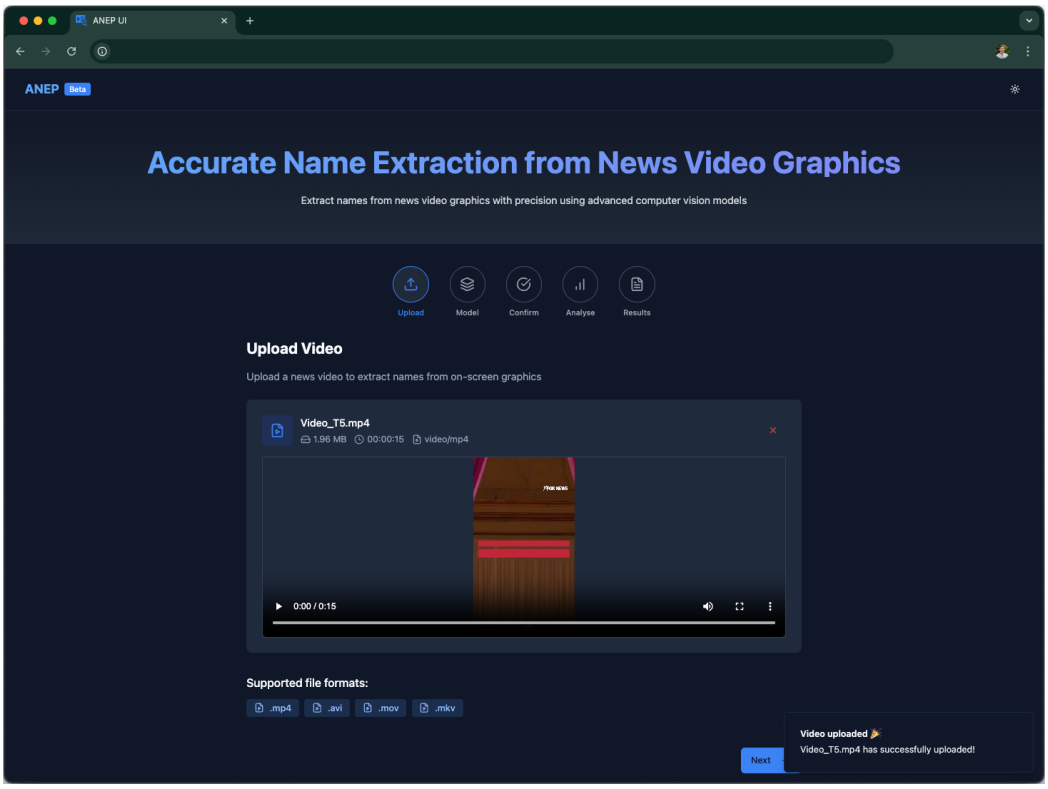


Figure G.4 A video file has been successfully uploaded to the ANEP platform. Metadata such as filename, size, duration, and format are displayed prior to proceeding to model selection

G.2.2 Model Selection Step

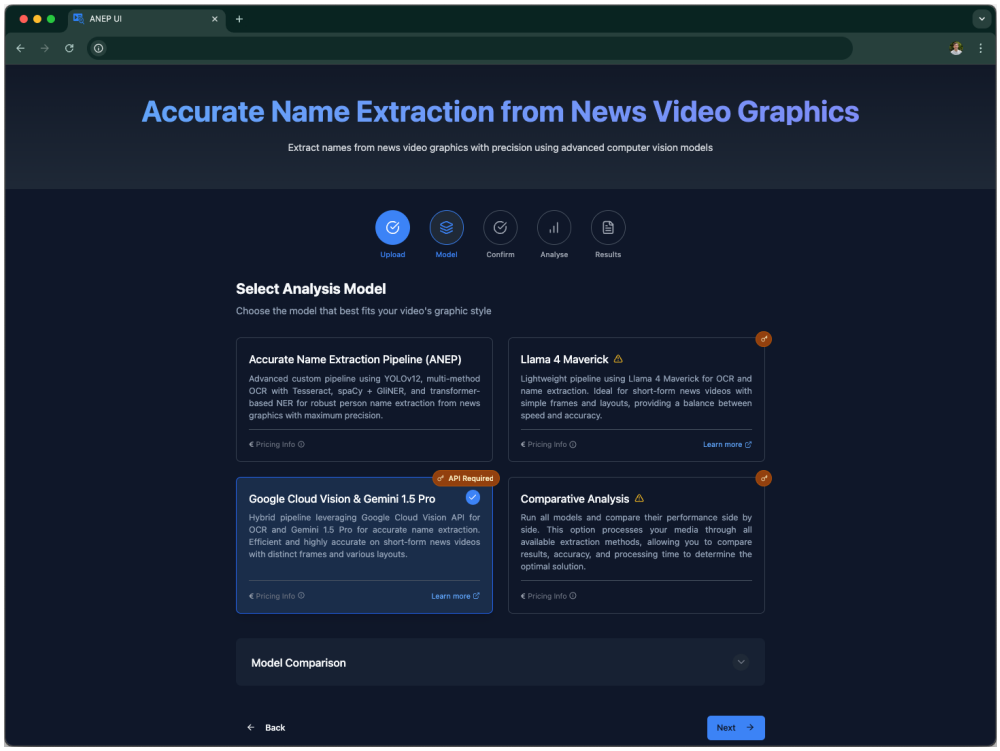


Figure G.5 Model selection interface with four pipelines: ANEP, LLaMA 4 Maverick, Google Cloud Vision & Gemini, and Comparative Analysis, each with a brief description

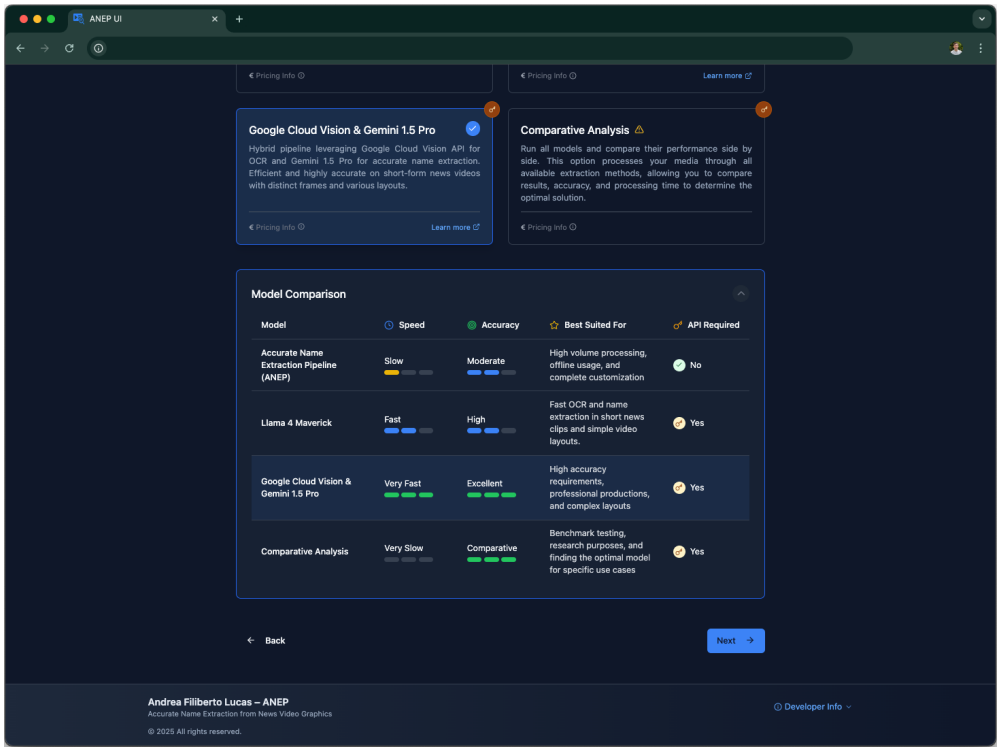


Figure G.6 Model comparison table showing the relative performance of different name extraction pipelines. Metrics include speed, accuracy, suitability, and API requirements

G.2.3 Confirm Step

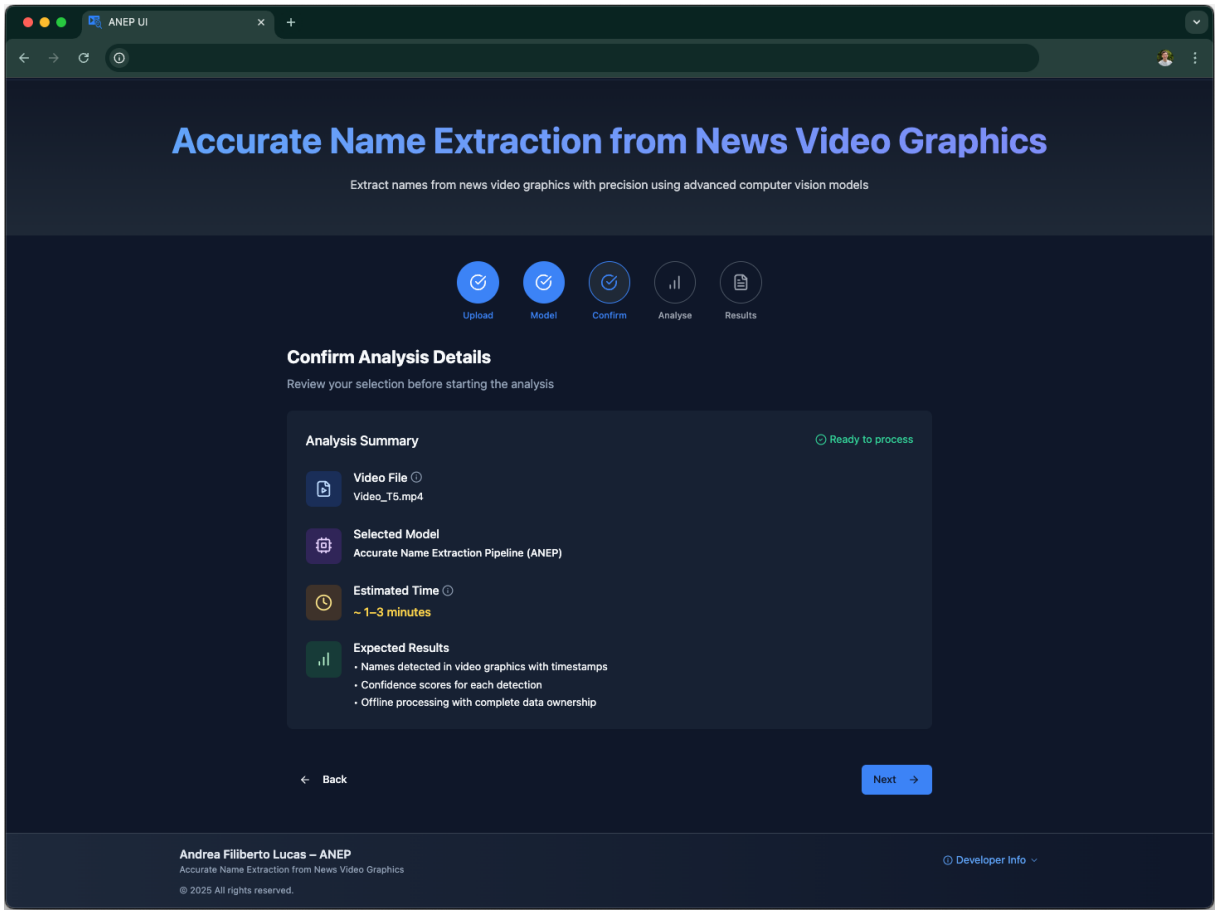


Figure G.7 Confirmation screen summarising selected analysis details, including video filename, chosen model, estimated processing time, and expected outputs

G.2.4 Analyse Step

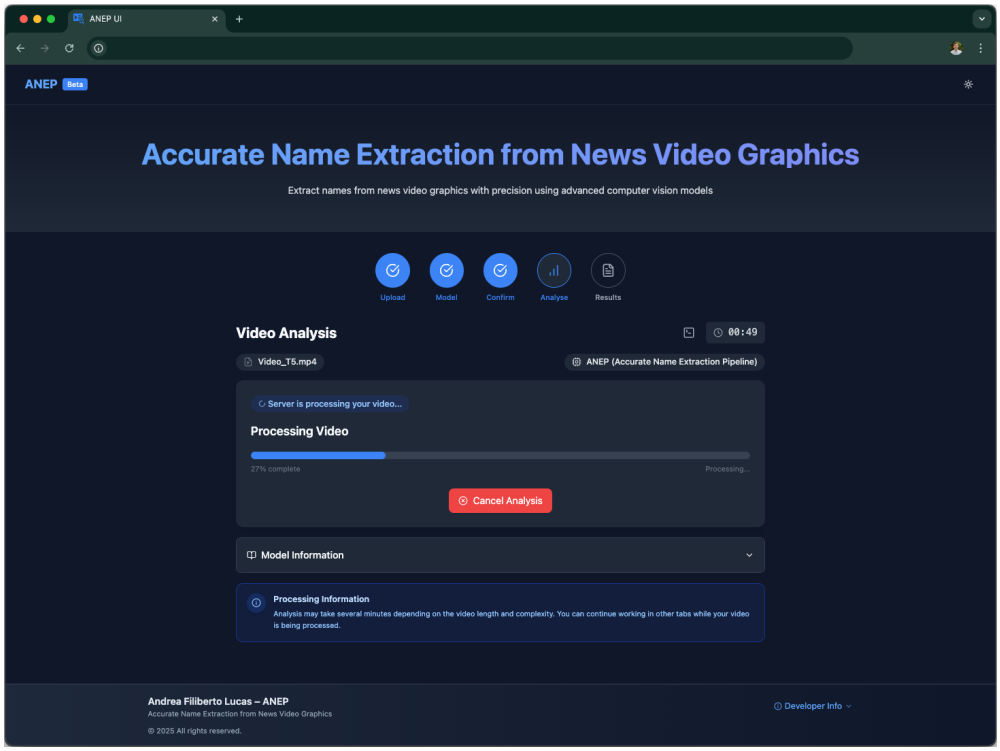


Figure G.8 The video analysis step in progress, showing the current completion percentage and processing time - Users may cancel the analysis at any stage

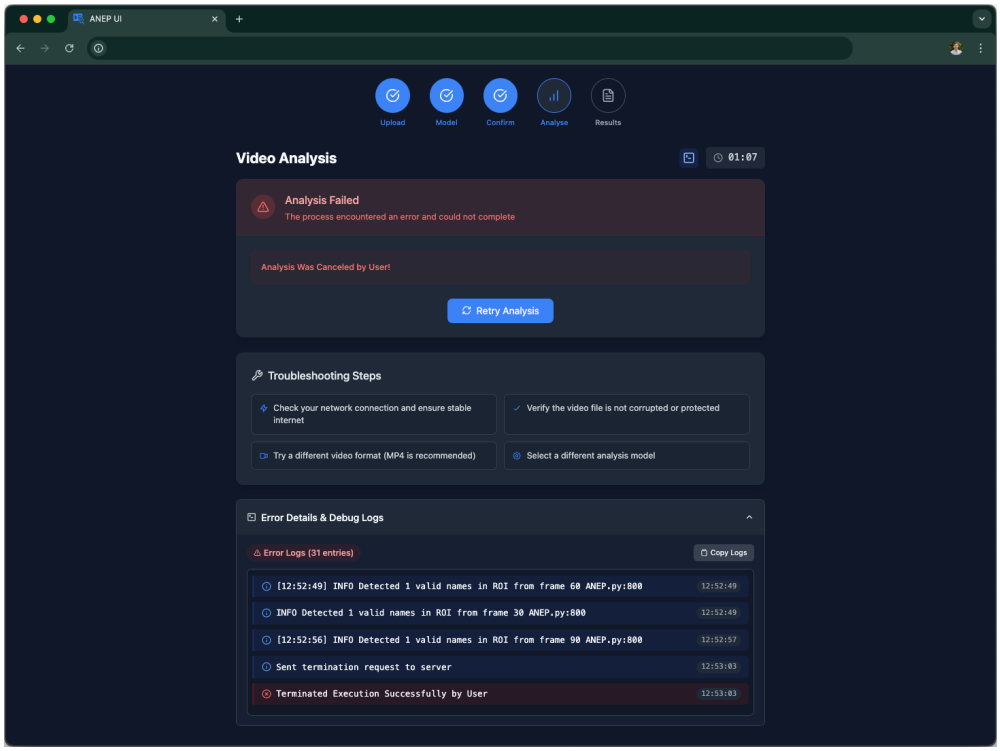


Figure G.9 The interface following an analysis cancellation - The UI displays an error state with logs and troubleshooting tips to help resolve common issues

G.2.5 Results Step

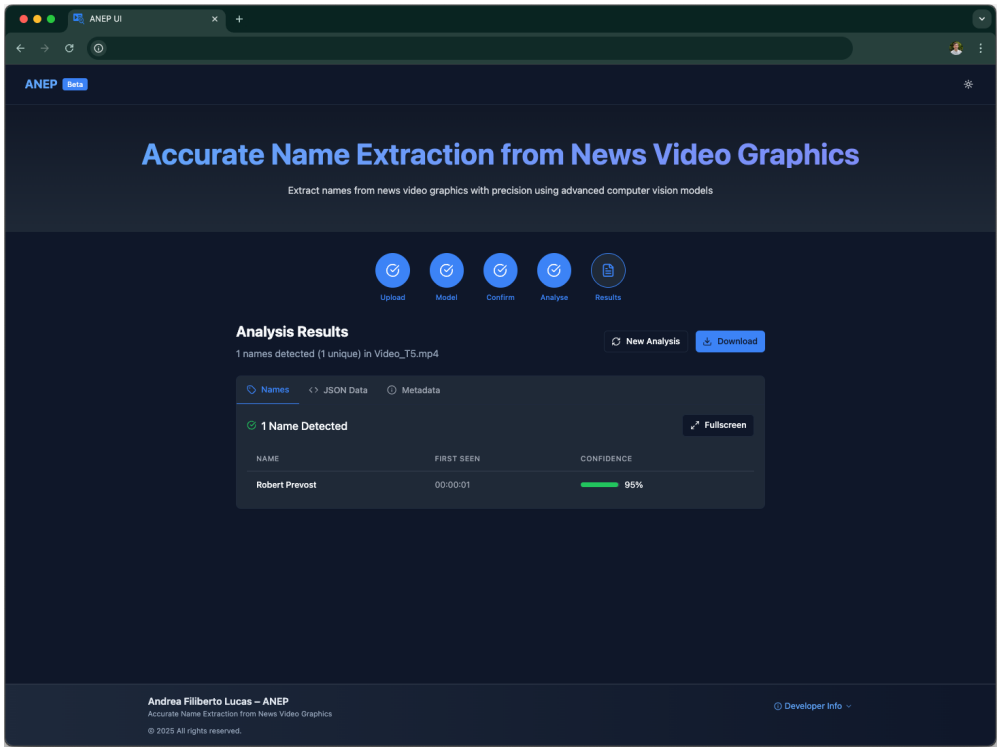


Figure G.10 Tabular view of detected names. Includes name, timestamp, and confidence score

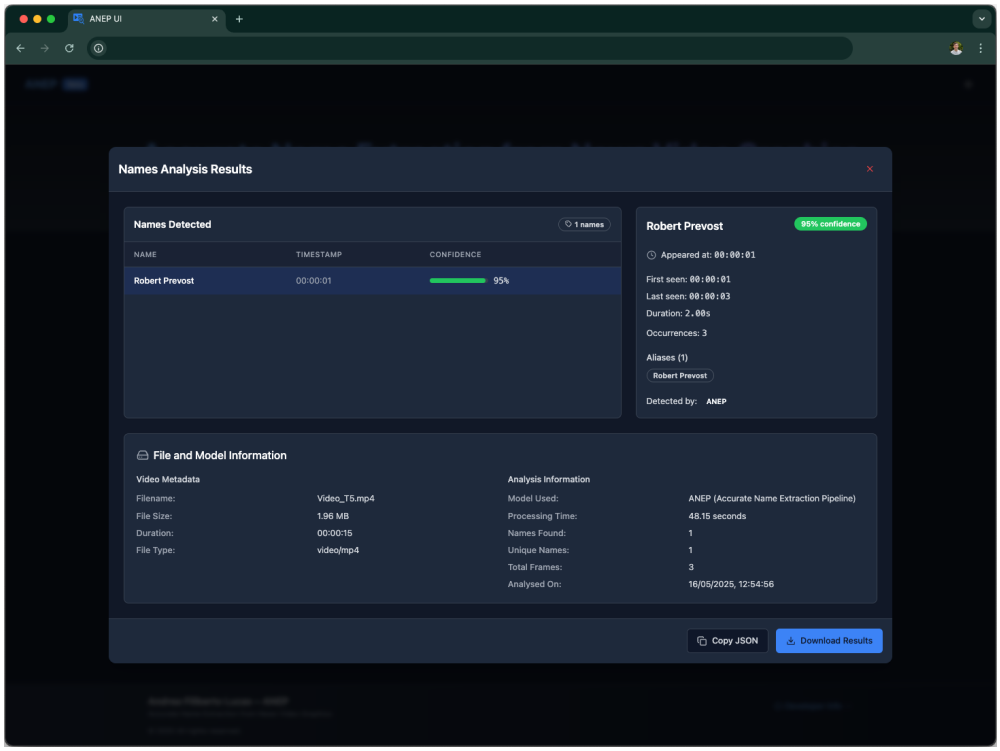


Figure G.11 Compact result view displaying the name detected, initial timestamp, and confidence - The UI offers a full-screen toggle for easier inspection

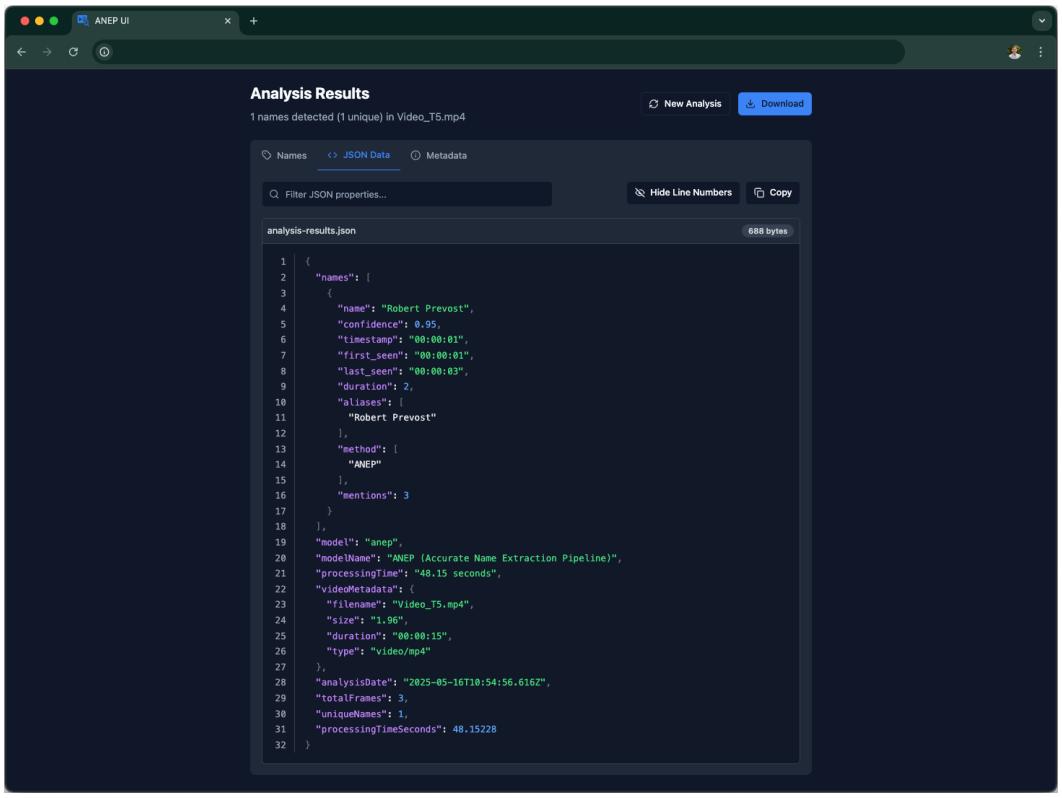


Figure G.12 Structured JSON view of the name detection results, including timing information, model metadata, and entity confidence scores

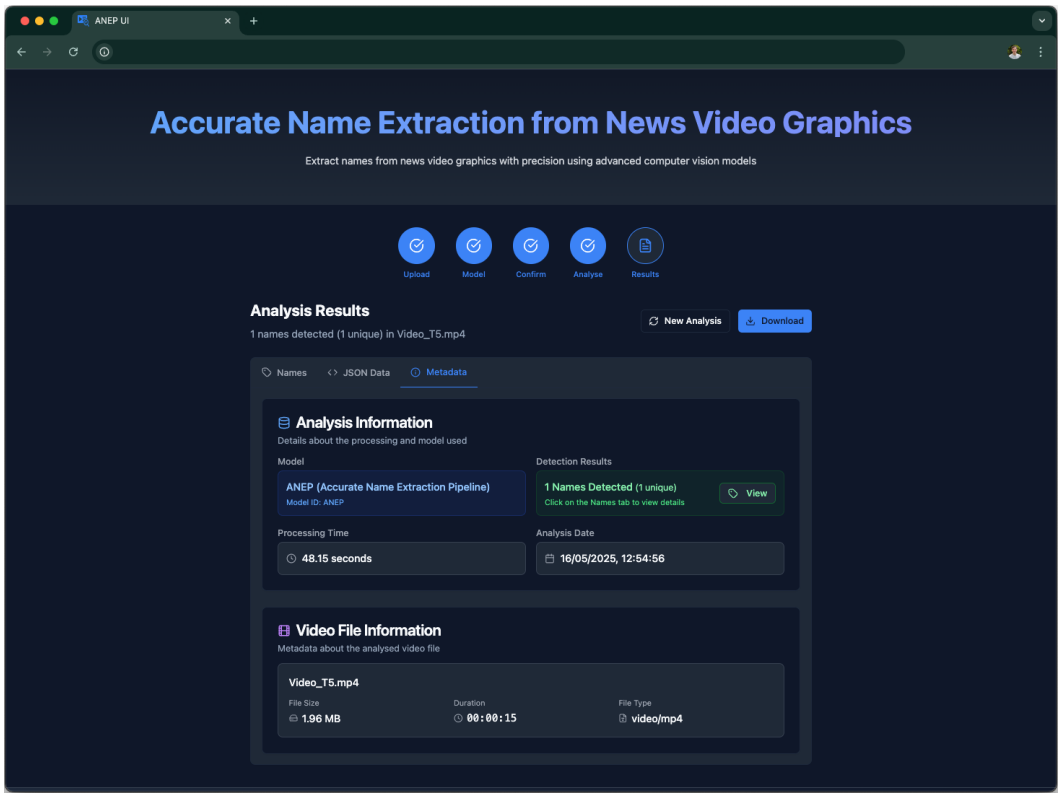


Figure G.13 Metadata tab showing model information, detection overview, and file metadata - Useful for exporting or archiving detection reports

G.2.6 Survey Visualisation

The survey visualisation interface equips users with interactive tools to explore public opinion data gathered through the integrated survey system. Users may select questions for analysis, apply demographic filters, and view responses using various chart types including bar, line, and pie formats. Export functionality is provided for downloading chart images or datasets in CSV format. Insights panels and settings controls enable thorough analysis, particularly when combined with filtering by age, gender, or residency.

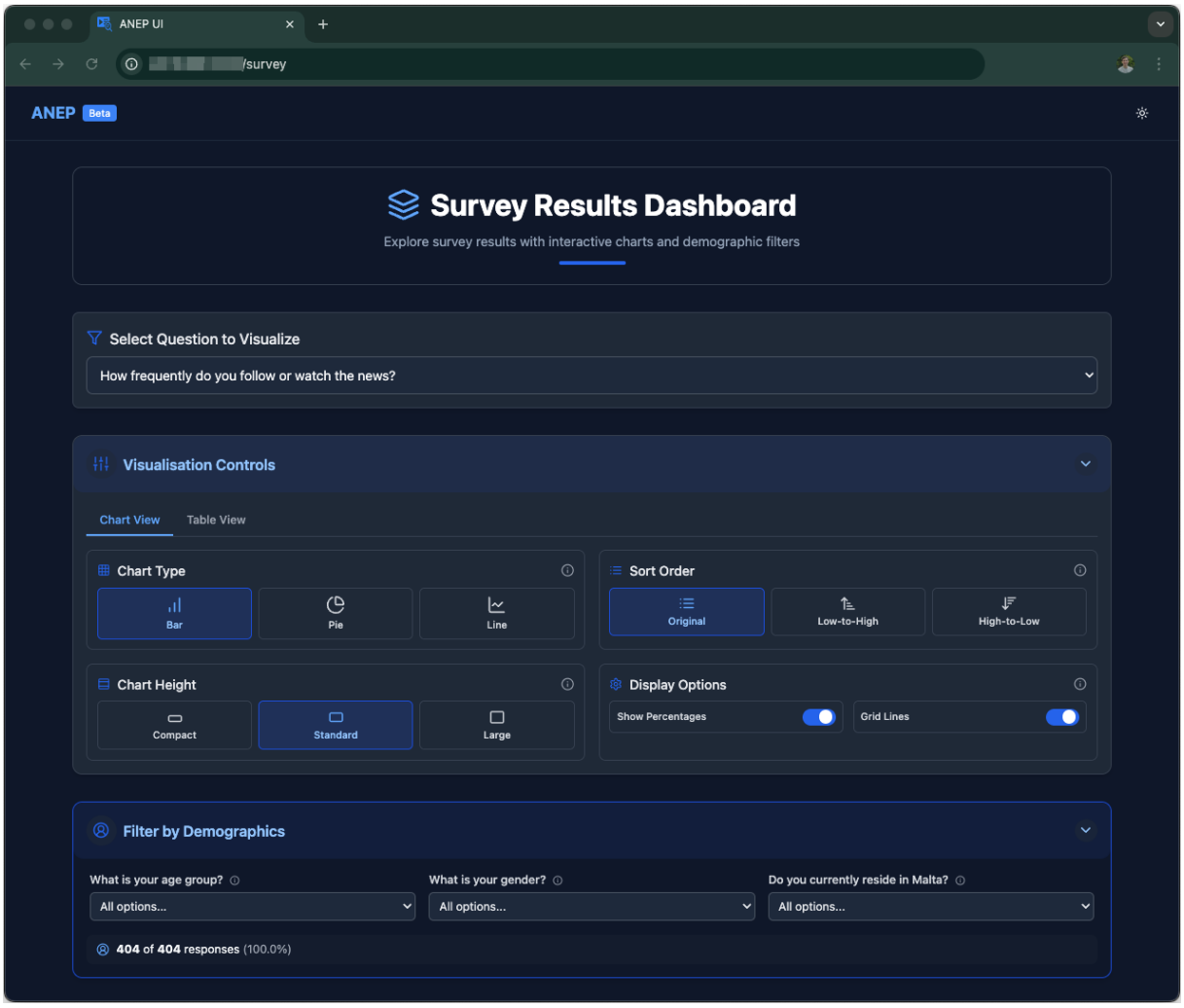


Figure G.14 Visualisation controls interface allowing users to toggle chart types, sort order, and display settings for survey data presentation

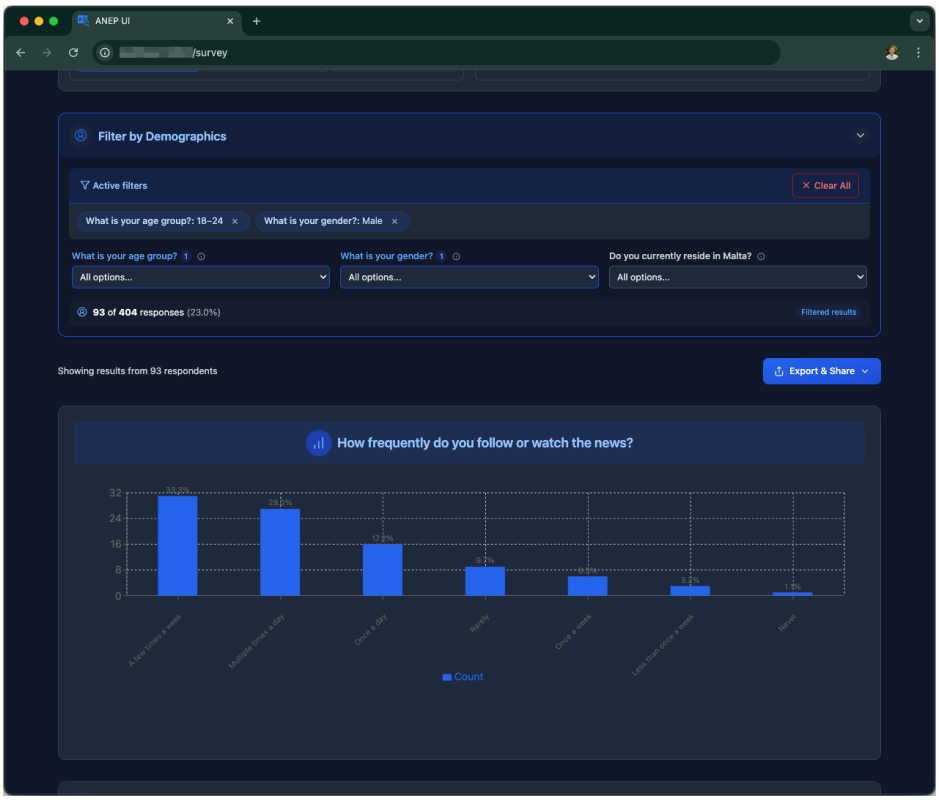


Figure G.15 Bar chart showing news consumption frequency for males aged 18-24, enabling demographic subgroup analysis

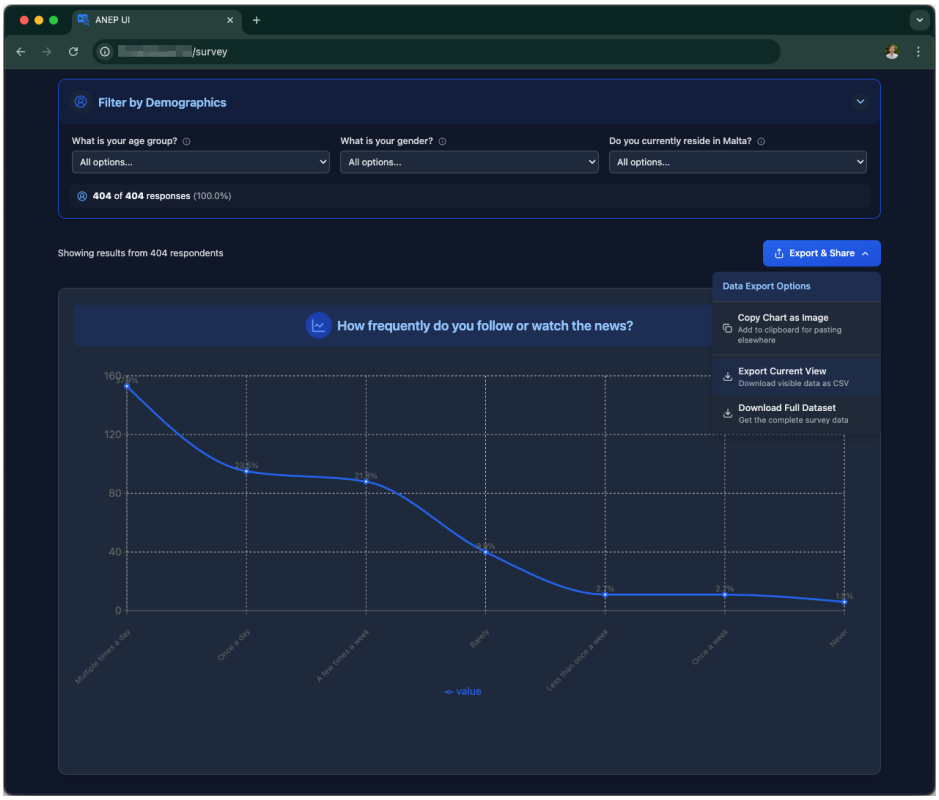


Figure G.16 Line chart of responses with export options for chart copying, filtered CSV data, and full dataset download

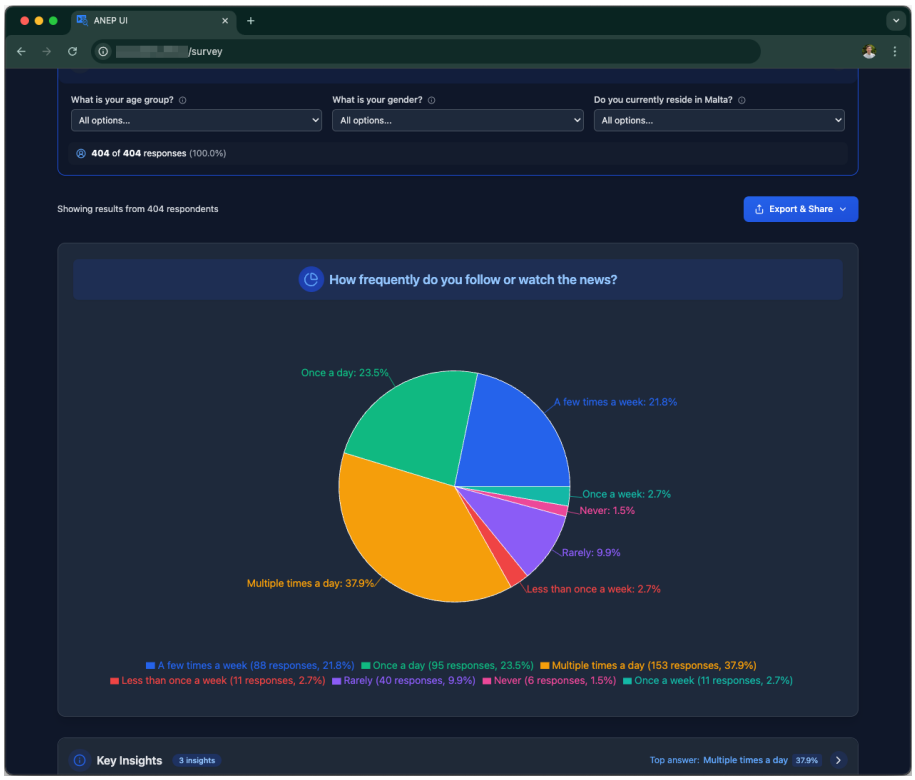


Figure G.17 Pie chart representation showing proportional responses to the question on news consumption frequency

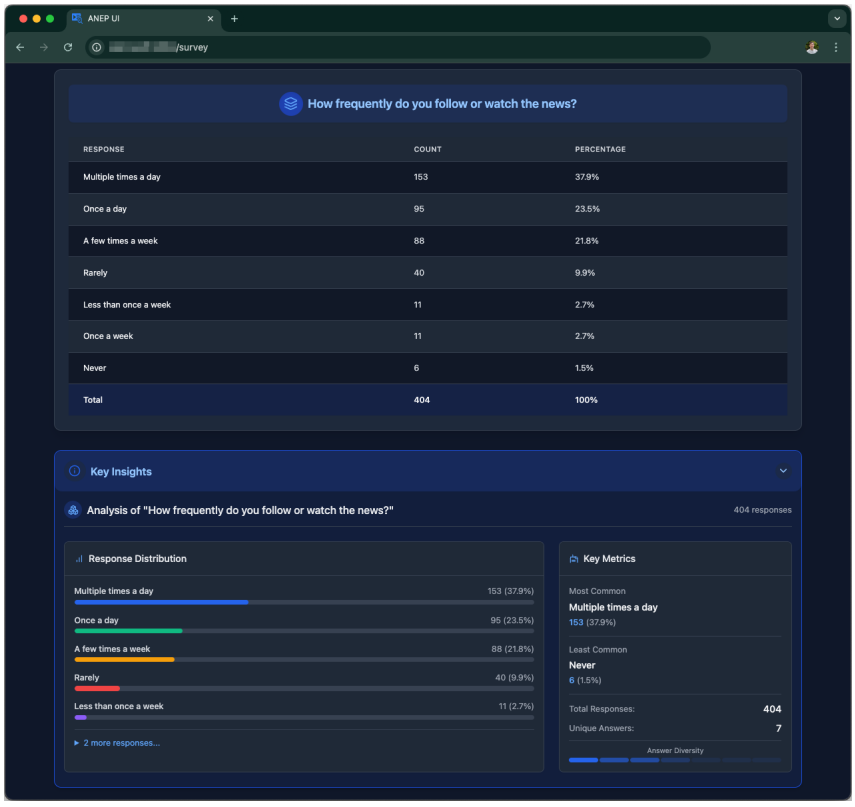


Figure G.18 Tabular survey result view with detailed response breakdown, count, and percentage