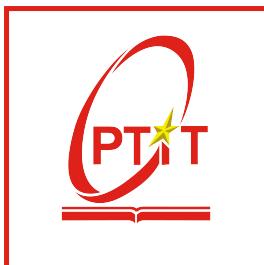


BỘ KHOA HỌC VÀ CÔNG NGHỆ
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



ĐỀ CƯƠNG

YOLOv13-based Food Detection and Calorie Estimation

Ứng dụng mô hình YOLOv13 trên tập dữ liệu ECUSTFD
cho phát hiện thực phẩm và ước lượng năng lượng

Giảng viên hướng dẫn: Kim Ngọc Bách

Sinh viên thực hiện: Đặng Văn Chiến

Mã sinh viên: B23DCCE012

Lớp: D23CQCE06-B

Niên khóa: 2023 – 2028

Hệ đào tạo: Đại học chính quy

Hà Nội, tháng 3 năm 2026

Danh mục các từ viết tắt

Từ viết tắt	Ý nghĩa
AI	<i>Artificial Intelligence</i> – Trí tuệ nhân tạo
AP	<i>Average Precision</i> – Độ chính xác trung bình
CNN	<i>Convolutional Neural Network</i> – Mạng nơ-ron tích chập
COCO	<i>Common Objects in Context</i> – Tập dữ liệu benchmark phát hiện đối tượng
CTDT	Chương trình đào tạo
ECUSTFD	<i>East China University of Science and Technology Food Dataset</i>
FPS	<i>Frames Per Second</i> – Số khung hình mỗi giây
GFLOPs	<i>Giga Floating Point Operations</i> – Đơn vị đo độ phức tạp tính toán
GVHD	Giảng viên hướng dẫn
MAE	<i>Mean Absolute Error</i> – Sai số tuyệt đối trung bình
mAP	<i>mean Average Precision</i> – Trung bình của AP
MRE	<i>Mean Relative Error</i> – Sai số tương đối trung bình
RMSE	<i>Root Mean Squared Error</i> – Căn bậc hai sai số bình phương trung bình
WHO	<i>World Health Organization</i> – Tổ chức Y tế Thế giới
YOLO	<i>You Only Look Once</i> – Họ mô hình phát hiện đối tượng thời gian thực

Mục lục

Danh mục các từ viết tắt	1
1 Giới thiệu	4
1.1 Lý do chọn đề tài	4
1.2 Mục tiêu nghiên cứu	4
1.2.1 Mục tiêu tổng quát	4
1.2.2 Mục tiêu cụ thể	4
1.2.3 Câu hỏi nghiên cứu	5
1.3 Ý nghĩa, tính ứng dụng và giá trị học thuật	5
1.4 Đối tượng và phạm vi nghiên cứu	5
1.5 Phương pháp nghiên cứu	6
2 Cơ sở lý thuyết và công nghệ sử dụng	7
2.1 YOLOv13: Kiến trúc và các cải tiến chính	7
2.1.1 Hạn chế của các phương pháp trước	7
2.1.2 HyperACE	7
2.1.3 FullPAD	7
2.1.4 Model Lightweighting	7
2.1.5 Kết quả trên tập COCO	8
2.2 ECUSTFD Dataset	8
2.2.1 Giới thiệu và lý do lựa chọn	8
2.2.2 Thống kê tập dữ liệu	9
2.3 Phương pháp ước lượng caloric hiện tại (Baseline)	9
2.4 Các công nghệ hỗ trợ	10
2.5 Tổng quan nghiên cứu liên quan	10
2.5.1 Nghiên cứu ngoài nước	10
2.5.2 Nghiên cứu trong nước	10
2.5.3 Khoảng trống nghiên cứu	11

3 Phân tích yêu cầu dự án	12
3.1 Mô tả tổng quan hệ thống	12
3.2 Yêu cầu phần nghiên cứu	12
3.2.1 Yêu cầu dữ liệu	12
3.2.2 Yêu cầu huấn luyện	12
3.2.3 Chỉ số đánh giá	13
3.3 Yêu cầu ứng dụng demo	13
4 Kế hoạch thực hiện	14
4.1 Tiến độ thực hiện	14
4.2 Kết quả dự kiến	15
4.3 Bố cục dự kiến báo cáo cuối kỳ	15

1. Giới thiệu

1.1 Lý do chọn đề tài

Béo phì là một trong những vấn đề sức khỏe công cộng nghiêm trọng nhất thế giới. Theo Tổ chức Y tế Thế giới (WHO), số người mắc béo phì đã tăng gấp ba lần kể từ năm 1975. Việc kiểm soát lượng *calorie* tiêu thụ hàng ngày là yêu cầu thiết yếu trong điều trị và phòng ngừa béo phì, tuy nhiên các phương pháp ghi chép thủ công hiện tại vẫn còn nhiều hạn chế: tốn thời gian, dễ quên, khó ước lượng chính xác, và đòi hỏi kiến thức dinh dưỡng chuyên sâu.

Trong bối cảnh đó, công nghệ thị giác máy tính (*Computer Vision*) mở ra hướng tiếp cận tự động hóa đầy hứa hẹn. **YOLOv13** – phiên bản mới nhất của họ mô hình YOLO, được công bố năm 2025 [1] – tích hợp hai cải tiến đột phá: **HyperACE** (mô hình hóa tương quan bậc cao dựa trên siêu đồ thị) và **FullPAD** (tích hợp toàn bộ *pipeline* đặc trưng). Song song đó, **ECUSTFD** là tập dữ liệu thực phẩm duy nhất hiện có đầy đủ thông tin *volume*, *mass*, *density* và *energy* cho từng loại – nền tảng không thể thiếu cho bài toán ước lượng *calorie*.

Tính đến thời điểm thực hiện đề tài, *chưa có nghiên cứu nào kết hợp YOLOv13 với ECUSTFD*. Đây chính là khoảng trống nghiên cứu mà đề tài hướng đến lấp đầy – thay thế Faster R-CNN (2015) [3] bằng công nghệ AI tiên tiến nhất năm 2026.

1.2 Mục tiêu nghiên cứu

1.2.1 Mục tiêu tổng quát

Ứng dụng mô hình YOLOv13 (với *HyperACE* và *FullPAD*) trên tập dữ liệu ECUSTFD để xây dựng hệ thống tự động phát hiện thực phẩm và ước lượng *calorie*, đạt hiệu suất vượt trội so với *baseline* Faster R-CNN [2], và phát triển ứng dụng *demo* minh họa chạy được trên máy tính cá nhân.

1.2.2 Mục tiêu cụ thể

Để đảm bảo tính khả thi trong phạm vi đề tài cá nhân **3 tháng** (28/02 – 31/05/2026), các mục tiêu được sắp xếp theo thứ tự ưu tiên:

1. **Nghiên cứu kiến trúc YOLOv13:** Tìm hiểu và giải thích cơ chế *HyperACE*, *FullPAD* và *model lightweighting*; so sánh với YOLOv12 và Faster R-CNN.
2. **Phân tích và chuẩn bị dữ liệu ECUSTFD:** Nắm cấu trúc 2.978 ảnh, 19 loại, 2 góc *view*; thực hiện *convert annotation* (PASCAL VOC → YOLO), *data augmentation* cơ bản, chia tập *train/val/test* (70%/15%/15%).

3. **Huấn luyện và đánh giá YOLOv13:** Huấn luyện 2 biến thể Nano và Small trên ECUSTFD; đánh giá qua *precision*, *recall*, F1, AP@0.5, mAP; so sánh với *baseline*.
4. **Xây dựng module ước lượng calorie:** Phát triển *pipeline* (YOLOv13 → GrabCut → ước lượng thể tích → $C = V \times D \times E$); kiểm tra độ chính xác qua MAE và MRE.
5. **Phát triển ứng dụng demo:** Giao diện *web* (Flask/FastAPI) cho phép *upload* ảnh, hiển thị kết quả phát hiện và ước lượng *calorie*.

1.2.3 Câu hỏi nghiên cứu

1. YOLOv13-N/S có cải thiện mAP, AP@0.5 trên ECUSTFD so với Faster R-CNN không? Mức cải thiện cụ thể là bao nhiêu?
2. Độ chính xác ước lượng *calorie* (MAE, MRE) có vượt qua mức $\pm 15\%$ của Liang & Li (2017) không?
3. Yếu tố nào (loại thực phẩm, góc chụp, ánh sáng) ảnh hưởng nhiều nhất đến sai số ước lượng?

1.3 Ý nghĩa, tính ứng dụng và giá trị học thuật

Giá trị khoa học: Đề tài lần đầu tiên áp dụng YOLOv13 trực tiếp trên ECUSTFD, tạo *benchmark* mới cho bài toán phát hiện thực phẩm và ước lượng *calorie*. Kết quả thực nghiệm sẽ cung cấp bằng chứng về hiệu quả thực tế của *HyperACE* và *FullPAD* trong miền dữ liệu thực phẩm.

Tính ứng dụng: Ứng dụng *demo* minh họa hệ thống tư vấn dinh dưỡng tự động – người dùng chỉ cần chụp ảnh bữa ăn và nhận ngay ước lượng *calorie*, giải quyết hạn chế của phương pháp ghi chép thủ công.

Giá trị học thuật: Đề tài rèn luyện đủ 4 năng lực theo chuẩn đầu ra học phần (CLO): tiếp cận công nghệ mới, tự học và giải quyết vấn đề, xây dựng ứng dụng *demo*, và viết báo cáo kỹ thuật chuẩn học thuật.

1.4 Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu:

- Mô hình YOLOv13-N và YOLOv13-S – hai biến thể nhẹ, cân bằng tốt tốc độ/độ chính xác, phù hợp tài nguyên cá nhân.
- Tập dữ liệu ECUSTFD: 2.978 ảnh, 19 loại, đầy đủ *volume/mass/density/energy records*, 2 góc chụp với đồng xu chuẩn hóa 25mm.
- *Pipeline* hoàn chỉnh: YOLOv13 → GrabCut → ước lượng thể tích → $C = V \times D \times E$.

Phạm vi kỹ thuật: Python 3.10+, PyTorch 2.0+, Ultralytics YOLOv13, OpenCV, Flask/FastAPI. Tối ưu hóa di động (ONNX/TFLite) nằm ngoài phạm vi chính.

Phạm vi thời gian: 28/02/2026 – 31/05/2026 (3 tháng; có thể kéo dài sang tháng 6 nếu cần, với sự đồng ý của GVHD).

1.5 Phương pháp nghiên cứu

Đề tài sử dụng **phương pháp thực nghiệm định lượng** gồm 4 bước:

1. **Thu thập và chuẩn bị dữ liệu:** Tải ECUSTFD, thực hiện *data augmentation* và *normalization* về 640×640 px.
2. **Huấn luyện và tối ưu mô hình:** Training YOLOv13-N/S, *batch size* 32–64, 300 *epochs* với *early stopping*, AdamW + cosine *LR decay*.
3. **Phát triển hệ thống và demo:** Xây dựng *pipeline* đầy đủ và giao diện người dùng.
4. **Đánh giá và phân tích:** So sánh với *baseline*; phân tích sai số theo loại và góc chụp.

2. Cơ sở lý thuyết và công nghệ sử dụng

2.1 YOLOv13: Kiến trúc và các cải tiến chính

2.1.1 Hạn chế của các phương pháp trước

Faster R-CNN [3] là *baseline* của nghiên cứu gốc ECUSTFD [2]: tốc độ chỉ 3–5 FPS, không đáp ứng *real-time*. Các phiên bản YOLO v1–v12 cải thiện tốc độ nhưng vẫn bị giới hạn bởi **tương quan theo cặp** (*pairwise correlation*) – chỉ so sánh vùng lân cận trực tiếp, không nắm bắt quan hệ toàn cục bậc cao.

2.1.2 HyperACE

HyperACE xây dựng **siêu đồ thị thích ứng** (*adaptive hypergraph*): mỗi vùng đặc trưng là một đỉnh; các *hyperedge* được học tự động để kết nối vùng có tương quan ngữ nghĩa cao dù không kề nhau về không gian. *Message passing* độ phức tạp tuyển tính tổng hợp thông tin toàn cục, cho phép nhận diện quan hệ *cross-location* và *cross-scale*.

Úng dụng với thực phẩm

HyperACE giúp mô hình phân biệt các loại trái cây hình dạng tương đồng (đào/táo/lê) dựa trên ngữ cảnh tổng thể, thay vì chỉ dựa vào đặc trưng cục bộ.

2.1.3 FullPAD

Thay vì luồng một chiều Backbone → Neck → Head, *FullPAD* tạo ra **3 đường hầm song song**:

1. *Tunnel 1*: Backbone → Neck (truyền thông tin chi tiết trực tiếp).
2. *Tunnel 2*: Bên trong Neck (tích hợp *multi-scale*).
3. *Tunnel 3*: Neck → Head (truyền đặc trưng tối ưu đến *head*).

2.1.4 Model Lightweighting

YOLOv13 thay thế tích chập nhân lớn bằng khối *DS-C3k/DS-C3k2*, giảm **30% tham số** và **28% GFLOPs** trong khi duy trì hoặc tăng độ chính xác.

2.1.5 Kết quả trên tập COCO

Mô hình	Params (M)	GFLOPs	mAP@0.5:0.9	Latency (ms)	vs. v12
YOLOv13-N	2.3	6.4	40.0%	1.97	+1.5%
YOLOv13-S	9.1	22.5	48.0%	3.56	+0.9%
YOLOv13-L	26.4	87.0	53.4%	8.49	+0.8%
Faster R-CNN	~41	~180	~37.0%	~200	Baseline

Bảng 2.1: Hiệu suất YOLOv13 trên MS COCO [1]

2.2 ECUSTFD Dataset

2.2.1 Giới thiệu và lý do lựa chọn

ECUSTFD [2] là **tập dữ liệu thực phẩm duy nhất** cung cấp đầy đủ *volume* và *mass records* thực đo cho từng mẫu. Tập dữ liệu sử dụng **đồng xu chuẩn hóa 25mm** để hiệu chuẩn kích thước và chụp theo **2 góc độ** (*top view + side view*) để tái tạo thể tích 3D từ ảnh 2D.

2.2.2 Thống kê tập dữ liệu

Loại thực phẩm	Số ảnh	Density	Energy	Hình dạng
Apple (Táo)	296	0.78	0.52	Ellipsoid
Banana (Chuối)	178	0.91	0.89	Cylinder cong
Bread (Bánh mì)	66	0.18	3.15	Hộp chũ nhật
Bun (Bánh bao)	90	0.34	2.23	Ellipsoid
Doughnut	210	0.31	4.34	Torus
Egg (Trứng)	104	1.03	1.43	Ellipsoid
Fried Doughnut	124	0.58	24.16	Cylinder
Grape (Nho)	58	0.97	0.69	Sphere
Lemon (Chanh)	148	0.96	0.29	Ellipsoid
Litchi (Vải)	78	1.00	0.66	Sphere
Mango (Xoài)	220	1.07	0.60	Ellipsoid
Mooncake (Bánh Trung Thu)	134	0.96	18.83	Cylinder dẹt
Orange (Cam)	254	0.90	0.63	Sphere
Peach (Đào)	126	0.96	0.57	Ellipsoid
Pear (Lê)	166	1.02	0.39	Ellipsoid lệch
Plum (Mận)	176	1.01	0.46	Sphere
Kiwi	120	0.97	0.61	Ellipsoid
Sachima	150	0.22	21.45	Hộp chũ nhật
Tomato (Cà chua)	172	0.98	0.27	Sphere
Mix (Hỗn hợp)	108	—	—	Không train
Tổng		2.978 ảnh, 19 loại chính		

Bảng 2.2: Thống kê chi tiết ECUSTFD [2] (Density: g/cm^3 ; Energy: $kcal/g$)

2.3 Phương pháp ước lượng calorie hiện tại (Baseline)

Liang & Li (2017) [2] đề xuất pipeline 4 bước:

1. Phát hiện đối tượng bằng Faster R-CNN: xác định loại thực phẩm và vị trí đóng xu.

2. Phân vùng bằng GrabCut [6]: trích xuất *contour* thực phẩm.
3. Uớc lượng thể tích từ *scale factor* (đồng xu 25mm) và công thức hình học.
4. Tính *calorie*: $C = V \times D \times E$

Kết quả: MRE $< \pm 15\%$ với hầu hết loại (ngoại trừ nho $\approx 33.5\%$). Tốc độ 3–5 FPS, không phù hợp *real-time*.

2.4 Các công nghệ hỗ trợ

- **Python 3.10+** – Ngôn ngữ lập trình chính.
- **PyTorch 2.0+** – Framework *deep learning* với GPU acceleration.
- **Ultralytics YOLOv13** – Thư viện chính thức YOLOv13.
- **OpenCV** – Xử lý ảnh, GrabCut *segmentation*.
- **NumPy, Pandas, Matplotlib** – Xử lý và trực quan hóa dữ liệu.
- **Flask/FastAPI** – Phát triển ứng dụng *demo web*.

2.5 Tổng quan nghiên cứu liên quan

2.5.1 Nghiên cứu ngoài nước

Lei et al. (2025) [1] công bố YOLOv13, chỉ đánh giá trên COCO – chưa có kết quả thực nghiệm nào trên dữ liệu thực phẩm hay ước lượng *calorie*.

Liang & Li (2017) [2] công bố ECUSTFD và *pipeline* ước lượng *calorie* nền tảng – nghiên cứu duy nhất sử dụng ECUSTFD, dùng công nghệ 2015–2017. Không có nghiên cứu nào cập nhật trên ECUSTFD với YOLO hiện đại trong 8 năm qua.

2.5.2 Nghiên cứu trong nước

Nhận diện món ăn Việt Nam: Một số nhóm tại ĐH Bách Khoa TP.HCM và ĐHQG TP.HCM thử nghiệm CNN/YOLOv5/v8 cho nhận diện phở, cơm tấm, bún bò – nhưng dừng ở bài toán phân loại, chưa xây dựng *pipeline* ước lượng *calorie*.

YOLO trong nông nghiệp và y tế: Một số luận văn dùng YOLOv5/v7/v8 phát hiện khuyết tật nông sản – chưa có ứng dụng YOLO từ v9 trở lên tại Việt Nam.

Ước lượng *calorie*: Chưa có nghiên cứu học thuật Việt Nam nào ước lượng *calorie* từ ảnh theo phương pháp *volume-based*.

2.5.3 Khoảng trống nghiên cứu

ECUSTFD chỉ được đánh giá với Faster R-CNN từ 2015. Trong 8 năm qua, không có nghiên cứu nào cập nhật *benchmark* trên ECUSTFD với YOLO hiện đại (v9–v13).

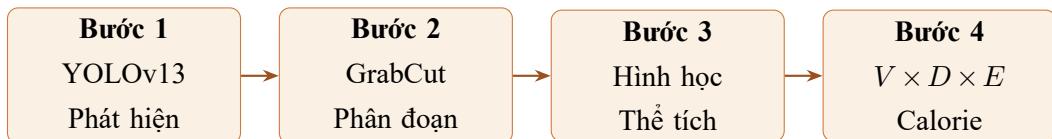
Bài báo gốc [1] chỉ đánh giá trên COCO. Hiệu quả thực tế của *HyperACE* và *FullPAD* trong miền dữ liệu thực phẩm chưa có bằng chứng thực nghiệm.

Chưa có hệ thống hoàn chỉnh từ ảnh đầu vào đến ước lượng *calorie* đầu ra được công bố tại Việt Nam.

Đề tài này được thiết kế để **lập đầy dòng thời cả ba khoảng trống** trên.

3. Phân tích yêu cầu dự án

3.1 Mô tả tổng quan hệ thống



Hình 3.1: Pipeline 4 bước của hệ thống

Bước	Module	Xử lý	Đầu ra
1	Phát hiện (YOLOv13)	Ảnh 2 góc, detect loại + đồng xu	{x,y,w,h,class,conf}
2	Phân đoạn (GrabCut)	Tách <i>foreground/background</i>	Mask + <i>contour</i>
3	Ước lượng thể tích	<i>Scale factor</i> + hình học	V (cm^3)
4	Tính <i>calorie</i>	Tra cứu D, E từ ECUSTFD	$C = V \times D \times E$ (kcal)

Bảng 3.1: Chi tiết pipeline 4 bước

3.2 Yêu cầu phần nghiên cứu

3.2.1 Yêu cầu dữ liệu

- ECUSTFD đầy đủ, chia *train/val/test* = 70%/15%/15%.
- *Convert*: PASCAL VOC (XML) → YOLO TXT.
- *Augmentation*: *rotation* ($\pm 15^\circ$), *scale* (0.8–1.2×), *brightness* (± 0.2), *flip*, *mosaic*.
- *Normalization*: 640×640 px.

3.2.2 Yêu cầu huấn luyện

- YOLOv13-N/S: *batch size* 32–64, 300 *epochs*, *early stopping* (*patience* = 20), AdamW + *cosine LR decay*.
- Đánh giá: *precision*, *recall*, F1, AP@0.5, mAP (19 *class*).
- So sánh với Faster R-CNN *baseline*.

3.2.3 Chỉ số đánh giá

Phát hiện:

$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN}, \quad F_1 = \frac{2PR}{P + R}$$

Ước lượng calorie:

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|, \quad \text{MRE} = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i}$$

3.3 Yêu cầu ứng dụng demo

- *Upload ảnh và hiển thị kết quả phát hiện (bounding boxes, labels, confidence).*
- Hiển thị ước lượng *calorie*: loại thực phẩm, V (cm^3), C (kcal).
- Giao diện trực quan, không yêu cầu kiến thức kỹ thuật.
- **Công nghệ ưu tiên:** Flask + HTML/CSS/JS.

4. Kế hoạch thực hiện

4.1 Tiến độ thực hiện

Đề tài bắt đầu **28/02/2026**, dự kiến hoàn thành **31/05/2026** (3 tháng):

Tháng	Tuần	Công việc cụ thể	Kết quả cần đạt
3	T1–T2	Đọc paper YOLOv13; tìm hiểu <i>HyperACE, FullPAD</i> ; cài đặt môi trường Python/PyTorch	Tóm tắt kỹ thuật (≥ 3 trang); <i>demo</i> YOLOv13 chạy được
	T3–T4	Đọc paper ECUSTFD; tải dataset; phân tích phân phối <i>class</i> , chất lượng ảnh	Báo cáo phân tích dữ liệu; thống kê <i>class imbalance</i>
4	T1–T2	<i>Convert annotation</i> ; chia tập; <i>augmentation</i> ; huấn luyện YOLOv13-N	Dataset sẵn sàng; kết quả sơ bộ YOLOv13-N trên <i>val set</i>
	T3–T4	Huấn luyện YOLOv13-S; đánh giá 2 mô hình trên <i>test set</i> ; so sánh <i>baseline</i>	Kết quả đầy đủ: mAP YOLOv13-N/S vs. Faster R-CNN
5	T1–T2	Module GrabCut + ước lượng thể tích + tính <i>calorie</i> ; test <i>pipeline</i> đầy đủ	<i>Pipeline</i> hoàn chỉnh; bảng MAE/MRE theo từng loại
	T3	Phát triển <i>demo</i> Flask; tích hợp toàn bộ <i>pipeline</i>	<i>Demo</i> chạy: <i>upload</i> ảnh → kết quả <i>calorie</i>
T4		Hoàn thiện báo cáo; nộp bản nháp GVHD; chỉnh sửa theo góp ý	Báo cáo hoàn chỉnh; nộp đúng hạn

Bảng 4.1: Kế hoạch tiến độ chi tiết (28/02/2026 – 31/05/2026)

Lưu ý về thời gian

Mốc 3 tháng là kế hoạch chính thức. Các hạng mục phức tạp (tối ưu *demo*, hoàn thiện báo cáo) có thể kéo dài sang tháng 6/2026 nếu cần, với sự đồng ý của GVHD.

4.2 Kết quả dự kiến

- **YOLOv13-N:** mAP@0.5 > 50% trên ECUSTFD (so với ~ 45% Faster R-CNN).
- **YOLOv13-S:** mAP@0.5 > 55% trên ECUSTFD.
- **Ước lượng calorie:** MRE < $\pm 15\%$; mục tiêu < $\pm 12\%$.
- **Tốc độ:** ≥ 15 FPS (CPU), ≥ 30 FPS (GPU) – vượt xa Faster R-CNN (3–5 FPS).
- **Demo:** Xử lý ảnh ≤ 3 giây trên máy không có GPU.
- **Báo cáo:** 36–45 trang, chuẩn học thuật, đầy đủ bảng số liệu và biểu đồ.

4.3 Bộ cục dự kiến báo cáo cuối kỳ

- **Lời mở đầu** (2–3 trang): Đặt vấn đề, lý do, mục tiêu.
- **Chương 1 – Cơ sở lý thuyết** (10–12 trang): YOLOv13 chi tiết, GrabCut, ước lượng thể tích.
- **Chương 2 – ECUSTFD & ước lượng calorie** (8–10 trang): Phân tích dataset, so sánh, *baseline*.
- **Chương 3 – Hệ thống đề xuất** (12–15 trang): *Pipeline* 4 bước, huấn luyện, *demo*.
- **Chương 4 – Kết quả & đánh giá** (10–12 trang): So sánh, phân tích sai số, thảo luận.
- **Kết luận** (2–3 trang) + **Tài liệu tham khảo** (2–3 trang).

Tài liệu tham khảo

- [1] M. Lei, S. Li, Y. Wu et al., “YOLOv13: Real-Time Object Detection with Hypergraph-Enhanced Adaptive Visual Perception,” *arXiv:2506.17733*, 2025.
- [2] Y. Liang and J. Li, “Computer Vision-based Food Calorie Estimation: Dataset, Method, and Experiment,” *arXiv:1705.07632v3*, 2017.
- [3] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” *NIPS*, 2015.
- [4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” *IEEE CVPR*, 2016.
- [5] L. Bossard, M. Guillaumin, and L. Van Gool, “Food-101: Mining Discriminative Components with Random Forests,” *ECCV*, 2014.
- [6] C. Rother, V. Kolmogorov, and A. Blake, “GrabCut: Interactive Foreground Extraction using Iterated Graph Cuts,” *ACM TOG*, 2004.
- [7] Y. Kawano and K. Yanai, “Automatic Expansion of a Food Image Dataset Leveraging Existing Categories with Domain Adaptation,” *ECCV Workshops*, 2015.
- [8] G. Ciocca, P. Napoletano, and R. Schettini, “Food Recognition: A New Dataset, Experiments, and Results,” *IEEE J. Biomed. Health Inform.*, vol. 21, no. 3, pp. 588–598, 2017.