

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA QUẢN LÝ DỰ ÁN



ĐỒ ÁN TỐT NGHIỆP

KHOA QUẢN LÝ DỰ ÁN
CHUYÊN NGÀNH QUẢN LÝ CÔNG NGHIỆP

ĐỀ TÀI

NGHIÊN CỨU ỨNG DỤNG MẠNG NƠ-RON TỰ MÃ HÓA (AUTOENCODER) TRONG PHÁT HIỆN KHUYẾT TẬT SẢN PHẨM VẢI

Giảng viên hướng dẫn: TS. HUỖNH NHẬT TỐ

Sinh viên thực hiện: NGUYỄN THỊ KỲ DUYÊN

Mã số sinh viên: 118210011

Lớp sinh hoạt: 21QLCN1

Đà Nẵng 2025



TÓM TẮT

Tên đề tài: Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải

Tóm tắt: Đề tài tập trung xây dựng hệ thống kiểm tra lỗi ngoại quan vải tự động bằng mô hình học sâu Autoencoder. Mục tiêu là hỗ trợ ra quyết định và tăng độ chính xác thay thế phương pháp kiểm tra thủ công hiện tại vốn phụ thuộc vào con người, dễ bỏ sót lỗi và hiệu quả thấp. Mô hình Autoencoder được huấn luyện không giám sát trên ảnh vải không lỗi vì ảnh về lỗi sản phẩm hiếm, đa dạng; Chi phí thu thập thấp; Thời gian label (gắn nhãn) vùng lỗi khi áp dụng học giám sát.

Ưu điểm của mô hình: Không cần dữ liệu lỗi khi huấn luyện, khả năng tái tạo ảnh chính xác, nhạy bén với sai lệch nhỏ, tốc độ xử lý cao và dễ mở rộng. Đề tài đề xuất tích hợp mô hình học sâu vào dây chuyền sản xuất thông qua hệ thống camera công nghiệp để tự động phát hiện và phân loại vải lỗi, từ đó tối ưu quy trình kiểm tra, giảm nhân lực, tăng năng suất và chất lượng sản phẩm.

Mô hình đạt $AUC = 0.88$, cho thấy hiệu quả phát hiện lỗi tốt và tiềm năng ứng dụng cao trong công nghiệp dệt may.

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Nguyễn Thị Kỳ Duyên Số thẻ sinh viên: 118210011

Lớp: 21QLCN1 Khoa: Quản lý Dự án Ngành: Quản lý công nghiệp

1. Tên đề tài đồ án:

Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải

2. Đề tài thuộc diện: ☐ Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện

3. Các số liệu và dữ liệu ban đầu:

- Quy trình kiểm tra chất lượng tại Công ty TNHH XDD Textile
- Số liệu thống kê lỗi tháng 11 đến tháng 4
- Tài liệu liên quan đến quản lý chất lượng và mô hình học máy

4. Nội dung các phần thuyết minh và tính toán:

- Giới thiệu mục tiêu
- Cơ sở lý thuyết
- Vấn đề doanh nghiệp
- Phân tích và thiết kế mô hình
- Đề xuất phương án cải tiến
- Đánh giá và hướng phát triển trong tương lai

5. Họ tên người hướng dẫn: TS. Huỳnh Nhật Tố

6. Ngày giao nhiệm vụ đồ án:

7. Ngày hoàn thành đồ án:

Đà Nẵng, ngày 16 tháng 06 năm 2025

Trưởng Bộ môn Quản lý Công nghiệp

Người hướng dẫn

TS. Huỳnh Nhật Tố

TS. Huỳnh Nhật Tố

LỜI CẢM ƠN

Sau một thời gian tìm hiểu và nghiên cứu đề tài “Ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải tại xưởng thành phẩm của Công ty TNHH XDD Textile”, em đã hoàn thành Đồ án tốt nghiệp đúng tiến độ đề ra. Kết quả này có được là nhờ vào sự đồng hành, hỗ trợ và động viên quý báu từ thầy cô, gia đình, bạn bè và các anh chị trong quá trình thực tập.

Trước hết, em xin bày tỏ lòng biết ơn sâu sắc đến Thầy TS.Huỳnh Nhật Tố, người đã tận tình hướng dẫn, luôn sẵn sàng chia sẻ kiến thức, kinh nghiệm và dành thời gian quý báu để giúp em hoàn thiện đồ án một cách tốt nhất. Sự đồng hành và góp ý của thầy là nguồn động lực lớn giúp em vững bước trong suốt quá trình thực hiện đề tài.

Em cũng xin chân thành cảm ơn Ban lãnh đạo Công ty TNHH XDD Textile đã tạo điều kiện cho em được thực tập, tiếp cận thực tế và nghiên cứu tại doanh nghiệp. Sự hỗ trợ nhiệt tình của quý công ty, đặc biệt là trong việc cung cấp tài liệu chuyên môn và giải đáp thắc mắc, đã góp phần quan trọng vào quá trình triển khai và hoàn thành đề tài.

Em xin gửi lời cảm ơn chân thành đến các anh, chị tại phòng thí nghiệm trung tâm và xưởng thành phẩm, những người đồng hành tuyệt vời luôn sẵn sàng giúp đỡ, chia sẻ kinh nghiệm thực tế trong suốt thời gian em thực tập. Đặc biệt, em xin được gửi lời tri ân đến chị Nguyễn Thị Diễm - Chủ quản Phòng thí nghiệm trung tâm, người đã luôn tận tâm hướng dẫn và hỗ trợ em bằng cả trách nhiệm và sự nhiệt huyết.

Luận văn đã có những kết quả cụ thể, nhưng do thời gian có hạn và kiến thức thực tế còn hạn chế, luận văn chắc chắn vẫn còn nhiều thiếu sót. Em rất mong nhận được sự đóng góp ý kiến từ quý thầy cô để có thể hoàn thiện hơn về nội dung và nâng cao kiến thức chuyên môn cho bản thân.

Cuối cùng, em xin kính chúc Thầy TS.Huỳnh Nhật Tố, Ban lãnh đạo Công ty TNHH XDD Textile cùng toàn thể các anh/chị sức khỏe dồi dào, nhiều may mắn và thành công trong công việc.

Em xin chân thành cảm ơn!

CAM ĐOAN

Sinh viên: Nguyễn Thị Kỳ Duyên, lớp: 21QLCN1 xin cam đoan

- Đồ án tốt nghiệp là thành quả của quá trình nghiên cứu học hỏi dựa trên cơ sở lý thuyết, số liệu thực tế thu thập được, thực hiện theo sự hướng dẫn của giảng viên hướng dẫn.
- Mọi sự tham khảo trong đồ án được trích nguồn và nằm trong danh mục tài liệu tham khảo.
- Nếu có sự sao chép kết quả của đề tài khác, em xin hoàn toàn chịu trách nhiệm.

Đà Nẵng, tháng 06 năm 2025

Sinh viên thực hiện

Ký tên

Nguyễn Thị Kỳ Duyên

MỤC LỤC

TÓM TẮT.....	i
NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP	ii
LỜI CẢM ƠN.....	iii
CAM ĐOAN.....	iv
MỤC LỤC	v
DANH MỤC HÌNH	x
DANH MỤC BẢNG	xii
DANH MỤC CÁC TỪ VIẾT TẮT.....	xiii
CHƯƠNG 1: GIỚI THIỆU MỤC TIÊU	1
1.1 Lý do chọn đề tài	1
1.2 Mục tiêu	2
1.3 Phạm vi nghiên cứu	2
1.4 Thực hiện phương pháp.....	2
1.5 Kết cấu của đồ án	2
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	4
2.1. Quản lý chất lượng	4
2.1.1. Khái niệm về quản lý chất lượng.....	4
2.1.2. Tầm quan trọng của quản lý chất lượng	4
2.1.3 Các chỉ số đo lường chất lượng phổ biến	5
2.2. Các công cụ truyền thống trong quản lý chất lượng	6
2.2.1 Phiếu kiểm tra (Check Sheet)	6

2.2.2 Biểu đồ Pareto (Pareto Chart).....	7
2.2.3 Biểu đồ nhân quả (Biểu đồ xương cá).....	8
2.2.4 Biểu đồ mật độ phân bố (Histogram)	9
2.2.5 Biểu đồ kiểm soát (Control Chart)	10
2.2.6 Lưu đồ (Flowchart).....	11
2.2.7 Biểu đồ phân tán (Scatter Diagram)	12
2.3. Hạn chế của công cụ truyền thống và xu hướng ứng dụng học máy trong kiểm tra chất lượng vải	12
2.3.1. Hạn chế của các công cụ truyền thống	12
2.3.2. Xu hướng ứng dụng học máy trong kiểm tra chất lượng vải	13
2.3.3 Các loại học máy	14
2.3.3.1 Học có giám sát (Supervised Learning)	14
2.3.3.2 Học không giám sát (Unsupervised Learning).....	14
2.4 Cơ sở lý thuyết về mô hình học máy	15
2.4.1 Giới thiệu về mô hình học máy	15
2.4.2 Các khái niệm cốt lõi hỗ trợ việc xây dựng và huấn luyện mô hình.....	16
2.4.3 Khái niệm chỉ số đánh giá	17
2.5 Giới thiệu về mô hình	19
2.5.1 Mô hình được sử dụng.....	19
2.5.2 Lí do chọn mô hình.....	20
2.5.3 Tính ưu việt của mô hình.....	21
2.6 Nghiên cứu liên quan.....	22
CHƯƠNG 3: VẤN ĐỀ DOANH NGHIỆP	25

3.1 Hiện trạng xử lý lỗi vãi ngoại quan	25
3.2 Lỗi thông thường	27
3.3 Dữ liệu thống kê tỉ lệ lỗi	30
CHƯƠNG 4: PHÂN TÍCH VÀ THIẾT KẾ MÔ HÌNH	32
4.1 Cơ sở định hướng thiết kế mô hình	32
4.1.1 Sơ đồ quy trình phát triển mô hình học máy	32
4.1.2 Cơ sở định hướng thiết kế mô hình	33
4.2 Thu thập dữ liệu	33
4.3 Phân tích dữ liệu đầu vào	35
4.4 Tiền xử lý dữ liệu	36
4.4.1 Tiền xử lý dữ liệu thô	36
4.4.2 Tiền xử lý dữ liệu trong mô hình	37
4.5 Sơ đồ hệ thống xử lý dữ liệu	38
4.6 Thiết kế kiến trúc mô hình	40
4.6.1 Kiến trúc tổng thể của mô hình	40
4.6.2 Tham số mô hình	43
4.7 Các công thức toán học liên quan	44
4.7.1 Tính kích thước đầu ra sau Conv/Deconv	44
4.7.2 Tính số lượng tham số từng lớp	44
4.7.3 Tính tổng số tham số huấn luyện toàn mô hình	45
4.8 Huấn luyện mô hình	45
4.8.1 Hàm mất mát sử dụng trong mô hình	45
4.8.1.1 MS-SSIM loss	45

4.8.1.2 Edge Loss	46
4.8.1.3 Texture loss	46
4.8.1.4 Pixel Loss	46
4.8.1.5 Loss kết hợp và công thức tổng quát.....	47
4.8.2 Nguyên lý hoạt động của mô hình.....	47
4.8.3 Cấu hình huấn luyện.....	48
4.8.4 Hiệu suất huấn luyện	52
4.9 Đánh giá mô hình	54
4.9.1 Đánh giá chất lượng tái tạo.....	54
4.9.2 Chọn ngưỡng Max Difference cho mô hình phát hiện lỗi.....	56
4.9.3 Đánh giá độ nhạy của mô hình	59
CHƯƠNG 5: ĐỀ XUẤT PHƯƠNG ÁN CẢI TIẾN	60
CHƯƠNG 6: ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN TRONG TƯƠNG LAI	62
6.1 Đánh giá tính khả thi	62
6.1.1 Tính khả thi về kỹ thuật.....	62
6.1.2 Tính khả thi về thời gian triển khai	62
6.1.3 Tính khả thi về nguồn lực.....	64
6.1.4 Khả năng mở rộng và ứng dụng	64
6.1.5 Hạn chế và rủi ro	64
6.2 Hướng phát triển trong tương lai.....	65
6.2.1 Tối ưu hiệu suất xử lý của mô hình.....	65
6.2.2 Nâng cấp mô hình để phân loại lỗi chi tiết.....	65
6.2.3 Mở rộng sang các khâu sản xuất khác.....	67

KẾT LUẬN	68
TÀI LIỆU THAM KHẢO	69
PHỤ LỤC	72

DANH MỤC HÌNH

Hình 2.1: Phiếu kiểm tra.....	6
Hình 2.2: Biểu đồ Pareto	7
Hình 2.3: Cấu trúc chung của biểu đồ xương cá	8
Hình 2.4: Biểu đồ Histogram.....	9
Hình 2.5: Biểu đồ kiểm soát	10
Hình 2.6 : Biểu đồ phân tán.....	12
Hình 3.1: Sơ đồ hiện trạng kiểm tra chất lượng ngoại quan tại XDD Textile	25
Hình 4.1: Sơ đồ quy trình phát triển mô hình học máy	32
Hình 4.2: Biểu đồ phân chia dữ liệu.....	33
Hình 4.3: Cấu hình thông số đầu vào cho quá trình huấn luyện	34
Hình 4.4: Sơ đồ phân tích dữ liệu đầu vào	35
Hình 4.5: Sơ đồ tiền xử lý dữ liệu thô	36
Hình 4.6: Nhận biết dữ liệu loại bỏ và dữ liệu đạt chuẩn.....	36
Hình 4.7: Sơ đồ tiền xử lý dữ liệu trong mô hình	37
Hình 4.8: Ảnh gốc và ảnh đã chuyển đổi về dạng xám (Grayscale)	37
Hình 4.9: Sơ đồ hệ thống xử lý dữ liệu Autoencoder.....	38
Hình 4.10: Sơ đồ hệ thống xử lý dữ liệu Autoencoder thực	38
Hình 4.11: Sơ đồ kiến trúc tổng thể của mô hình.....	40
Hình 4.12: Cấu trúc Autoencoder với các lớp Conv2D và ConvTranspose2D	41
Hình 4.13: Tham số mô hình.....	43
Hình 4.14: Cấu hình huấn luyện.....	48
Hình 4.15: Trọng số huấn luyện	49
Hình 4.16: Tối ưu hóa AdamW và điều chỉnh tốc độ học.....	51
Hình 4.17: Huấn luyện mô hình	52

Hình 4.18: Hiệu suất huấn luyện	52
Hình 4.19: Thời gian xử lý ảnh của mô hình.....	53
Hình 4.20: Kiểm tra chất lượng tái tạo	54
Hình 4.21: Kiểm tra chất lượng tái tạo	54
Hình 4.22: Kiểm tra chất lượng tái tạo	55
Hình 4.23: Kiểm tra chất lượng tái tạo	55
Hình 4.24: Kiểm tra chất lượng tái tạo	56
Hình 4.25: Phân bố Max Difference giữa ảnh lỗi và ảnh không lỗi.....	57
Hình 4.26: Phân bố Max Difference giữa ảnh lỗi và ảnh không lỗi.....	58
Hình 4.27: Hiệu suất phân loại của mô hình	59
Hình 5.1: Sơ đồ phương án cải tiến quy trình kiểm tra chất lượng tại xưởng thành phẩm	60

DANH MỤC BẢNG

Bảng 2.1: Ký hiệu cơ bản được quy định trong Flowchart	11
Bảng 2.2: So sánh tính năng giữa chỉ số SSIM và Max Difference	19
Bảng 3.1: Các lỗi thường gặp	27
Bảng 3.2: Dữ liệu thống kê lỗi	30
Bảng 4.1: Thông tin dữ liệu	34
Bảng 4.2: Đánh giá chất lượng ảnh	34
Bảng 4.3: Trọng số của từng thành phần	49
Bảng 4.4: Ngưỡng phân loại lỗi theo Max Difference	58
Bảng 6.1: Kế hoạch triển khai hệ thống	63
Bảng 6.2: Hạn chế và rủi ro khi triển khai mô hình	64
Bảng 6.3: Phân tích dữ liệu lỗi	66
Bảng 6.4: Vấn đề và nghi vấn khi phân tích dữ liệu	66

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Tiếng anh	Giải thích
RQM	Total Quality Management	Quản lý chất lượng toàn diện
FPY	First Pass Yield	Tỷ lệ đạt lần đầu
HACCP	Hazard Analysis and Critical Control Points	Phân tích mối nguy và kiểm soát điểm tới hạn
ANSI	American National Standards Institute	Viện Tiêu chuẩn Quốc gia Hoa Kỳ
QC	Quality Control	Bộ phận kiểm soát chất lượng
AI	Artificial Intelligence	Trí tuệ nhân tạo
CNN	Convolutional Neural Network	Mạng nơ-ron tích chập
SVM	Support Vector Machine	Máy vector hỗ trợ
PCA	Principal Component Analysis	Phân tích thành phần chính
ICA	Independent Component Analysis	Phân tích thành phần độc lập
RNN	Recurrent Neural Network	Mạng nơ-ron hồi tiếp
LSTM	Long Short-Term Memory	Bộ nhớ ngắn dài hạn
BERT	Bidirectional Encoder Representations from Transformers	Biểu diễn mã hóa hai chiều từ mô hình Transformer
GPT	Generative Pre-trained Transformer	Transformer được huấn luyện trước để tạo sinh văn bản
GAN	Generative Adversarial Network	Mạng đối nghịch tạo sinh
MSE	Mean Squared Error	Sai số bình phương trung bình

PSNR	Peak Signal-to-Noise Ratio	Mức độ tương đồng giữa ảnh gốc và ảnh tái tạo
AE	Autoencoder	Bộ tự mã hóa
VAE	Variational Autoencoder	Bộ tự mã hóa biến phân
ReLU	Rectified Linear Unit	Hàm tuyến tính có chỉnh sửa
SSIM	Structural Similarity Index Measure	Chỉ số đo độ tương đồng cấu trúc
GPU	Graphics Processing Unit	Bộ xử lý đồ họa
CPU	Central Processing Unit	Bộ xử lý trung tâm
ROC	Receiver Operating Characteristic	Đặc trưng vận hành của bộ thu
AUC	Area Under the Curve	Diện tích dưới đường cong
Adam	Adaptive Moment Estimation	Ước lượng động lượng thích nghi
AdamW	Adam with Weight Decay	Adam kết hợp với suy giảm trọng số
CFD	Computational Fluid Dynamics	Tính toán động lực học chất lưu
EDP	Electronic Data Processing	Xử lý dữ liệu điện tử
QoI	Quantity of Interest	Đại lượng cần theo dõi
Conv2D	2D Convolution	Tích chập 2 chiều
ConvT2D	2D Transposed Convolution	Tích chập chuyển vị 2 chiều
DAGMM	Deep Autoencoding Gaussian Mixture Model	Mô hình phân phối hỗn hợp Gaussian kết hợp tự mã hóa sâu

CHƯƠNG 1: GIỚI THIỆU MỤC TIÊU

1.1 Lý do chọn đề tài

Ngày nay, với sự phát triển mạnh mẽ của thị giác máy tính (Computer Vision) và học sâu (Deep Learning), các ứng dụng tự động hóa trong công nghiệp, đặc biệt trong công đoạn kiểm tra chất lượng sản phẩm ngày càng được chú trọng. Ngành dệt may cũng không ngoại lệ, khi chất lượng bề mặt vải đóng vai trò then chốt quyết định giá trị sản phẩm cũng như uy tín thương hiệu của doanh nghiệp trên thị trường.

Tuy nhiên, tại nhiều công ty sản xuất vải hiện nay, quy trình kiểm tra lỗi ngoại quan vẫn chủ yếu được thực hiện thủ công, tức là công nhân trực tiếp quan sát bằng mắt thường để phát hiện các lỗi như tạp chất, sợi thô, đứt sợi, kết đầu,...

Phương pháp này tồn tại nhiều hạn chế nghiêm trọng [1]:

- Phụ thuộc lớn vào kinh nghiệm, sức khỏe và mức độ tập trung của người kiểm tra.
- Khả năng bỏ sót lỗi cao, đặc biệt với những lỗi nhỏ hoặc lỗi khó phát hiện bằng mắt thường.
- Thời gian kiểm tra kéo dài, yêu cầu nhiều nhân sự, dẫn đến tăng chi phí vận hành.

Trong thực tế sản xuất, một cuộn vải sẽ bị đánh giá không đạt chất lượng nếu số lượng lỗi vượt ngưỡng cho phép. Trong 1 mét không được có quá 4 lỗi. Trung bình mỗi tháng có 3 - 5 đơn hàng bị khiếu nại vì vải không đạt chất lượng như yêu cầu. Bởi nguyên nhân kiểm tra chất lượng vải không kĩ, từ đó dẫn đến hệ lụy hàng trăm thậm chí là hàng ngàn mét vải bị loại bỏ. Quy đổi theo giá trị kinh tế, doanh nghiệp có thể thiệt hại từ vài trăm triệu đến hàng tỷ đồng mỗi tháng và con số này tăng lên đáng kể trong một năm. Những thiệt hại này không chỉ ảnh hưởng đến lợi nhuận mà còn gây gián đoạn kế hoạch sản xuất, làm giảm uy tín doanh nghiệp và nguy cơ mất các đơn hàng lớn.

Việc kiểm soát chất lượng không hiệu quả kéo theo hàng loạt hệ lụy: hao hụt tài sản, tăng chi phí nhân công, giảm tốc độ giao hàng, tăng nguy cơ vi phạm cam kết với khách hàng. Do đó, việc tìm ra một giải pháp kiểm tra chất lượng vải tự động, nhanh chóng, chính xác và tiết kiệm chi phí là yêu cầu cấp thiết đối với các doanh nghiệp dệt may hiện nay.

Xuất phát từ thực tiễn đó, đề tài "Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải" được lựa chọn.

1.2 Mục tiêu

Với kỳ vọng xây dựng một hệ thống kiểm tra lỗi ngoại quan tự động, đề tài áp dụng mô hình học máy vào quá trình kiểm tra chất lượng ngoại quan của vải nhằm thay thế phương pháp kiểm tra thủ công còn nhiều hạn chế. Hệ thống này không chỉ giúp giảm thiểu tối đa sai sót do yếu tố con người gây ra, mà còn rút ngắn đáng kể thời gian kiểm tra, giảm thiểu chi phí nhân sự và tăng khả năng kiểm tra liên tục trong môi trường sản xuất công nghiệp. Đồng thời, việc tự động hóa quy trình kiểm tra chất lượng còn góp phần nâng cao năng suất, đảm bảo độ đồng đều trong sản phẩm đầu ra, từ đó cải thiện chất lượng tổng thể, tăng tính cạnh tranh và thúc đẩy quá trình chuyển đổi số của doanh nghiệp trong bối cảnh cuộc cách mạng công nghiệp 4.0 đang diễn ra mạnh mẽ.

1.3 Phạm vi nghiên cứu

Đề tài sẽ được triển khai nghiên cứu thực tế tại xưởng thành phẩm của Công ty TNHH XDD Textile, cụ thể là tại khu vực kiểm tra lỗi ngoại quan của vải - nơi diễn ra quá trình kiểm tra chất lượng sản phẩm trước khi đóng gói và xuất xưởng.

1.4 Thực hiện phương pháp

Trong quá trình thực hiện đề tài, nghiên cứu tiến hành quan sát trực tiếp quy trình kiểm tra lỗi ngoại quan do công nhân thực hiện tại xưởng, đồng thời thu thập dữ liệu hình ảnh các mẫu vải bị lỗi nhằm phục vụ cho việc xây dựng và huấn luyện mô hình. Song song đó, cũng thực hiện phỏng vấn một số công nhân kiểm vải để ghi nhận các loại lỗi thường gặp, đánh giá mức độ ảnh hưởng của lỗi đến chất lượng sản phẩm, cũng như thu thập ý kiến về hiệu quả và những khó khăn trong quá trình kiểm tra thủ công hiện nay.

1.5 Kết cấu của đồ án

Chương 1: Giới thiệu mục tiêu

Chương 2: Cơ sở lý thuyết

Chương 3: Vấn đề doanh nghiệp

Chương 4: Phân tích và thiết kế mô hình

Chương 5: Đề xuất phương án cải tiến

Chương 6: Đánh giá và hướng phát triển trong tương lai

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Quản lý chất lượng

2.1.1. Khái niệm về quản lý chất lượng

Quản lý chất lượng là tập hợp các hoạt động được thực hiện nhằm kiểm soát, duy trì và nâng cao chất lượng của sản phẩm hoặc dịch vụ. Trong sản xuất công nghiệp, đặc biệt là ngành dệt may, quản lý chất lượng đảm bảo rằng sản phẩm đầu ra đáp ứng các tiêu chuẩn kỹ thuật, thẩm mỹ và yêu cầu của khách hàng.

Quản lý chất lượng không chỉ tập trung vào việc kiểm tra sản phẩm sau cùng, mà còn bao gồm việc giám sát toàn bộ quy trình sản xuất từ đầu vào đến đầu ra nhằm ngăn ngừa lỗi phát sinh, thay vì chỉ phát hiện lỗi sau khi đã xảy ra [2].

2.1.2. Tầm quan trọng của quản lý chất lượng

Quản lý chất lượng đóng vai trò cốt lõi trong việc đảm bảo rằng sản phẩm hoặc dịch vụ đáp ứng kỳ vọng của khách hàng và phù hợp với các tiêu chuẩn quy định. Đặc biệt trong môi trường sản xuất công nghiệp và dệt may, quản lý chất lượng không chỉ là công cụ giám sát mà còn là một chiến lược toàn diện nhằm nâng cao hiệu quả vận hành, giảm thiểu lãng phí và tối ưu chi phí [2].

Việc triển khai hệ thống quản lý chất lượng hiệu quả giúp doanh nghiệp:

- Ngăn chặn lỗi từ sớm thay vì chỉ sửa lỗi ở đầu ra.
- Tăng năng suất và giảm thiểu phế phẩm, nhờ kiểm soát chặt chẽ quy trình.
- Gia tăng sự hài lòng và lòng trung thành của khách hàng, thông qua việc duy trì chất lượng ổn định.
- Nâng cao uy tín thương hiệu và khả năng cạnh tranh trên thị trường toàn cầu.

Theo *Eci Solutions (2023)*, các doanh nghiệp áp dụng quản lý chất lượng toàn diện có thể tăng lợi nhuận lên đến 26% và doanh thu trung bình thêm 9%, đồng thời giảm đáng kể số lượng sản phẩm lỗi trước khi đóng gói [3].

Trong ngành dệt may, việc áp dụng kiểm soát chất lượng trong từng công đoạn như cắt, may, hoàn thiện... giúp doanh nghiệp phát hiện lỗi sớm, từ đó giảm tỷ lệ hoàn trả, đồng thời rút ngắn thời gian kiểm tra thủ công, điều vốn tiêu tốn nhiều nhân lực [2].

Thực tế cũng cho thấy rằng việc áp dụng quản lý chất lượng hiện đại như TQM (Total Quality Management) và các công cụ kiểm soát thống kê đã cải thiện đáng kể hiệu suất vận hành trong ngành dệt, đặc biệt tại các doanh nghiệp đang theo đuổi sản xuất tinh gọn và chuyển đổi số [4].

2.1.3 Các chỉ số đo lường chất lượng phổ biến

Trong quá trình quản lý chất lượng, các doanh nghiệp thường sử dụng một số chỉ số đo lường (quality metrics) để theo dõi và đánh giá hiệu quả sản xuất:

- *Defect Rate (Tỷ lệ lỗi)*: Phản ánh hiệu suất kiểm soát chất lượng đầu ra. Giúp xác định mức độ ổn định của quy trình sản xuất.

$$\text{Tỷ lệ lỗi} = \frac{\text{Số sản phẩm lỗi}}{\text{Tổng số sản phẩm}} \times 100\% \quad (2.1)$$

- *First Pass Yield (FPY)*: Tỷ lệ sản phẩm đạt yêu cầu mà không cần kiểm tra lại hay sửa chữa.

$$FPY = \frac{\text{Sản phẩm đạt yêu cầu ngay lần đầu}}{\text{Tổng số sản phẩm đầu ra}} \times 100\% \quad (2.2)$$

- *Return Rate (Tỷ lệ hoàn trả)*: Tỷ lệ khách hàng trả lại sản phẩm hoặc phản ánh lỗi sau khi nhận hàng.

$$\text{Tỷ lệ hoàn trả} = \frac{\text{Sản phẩm bị khách khải lại}}{\text{Tổng số sản phẩm bán ra}} \times 100\% \quad (2.3)$$

- *Scrap Rate (Tỷ lệ phế phẩm)*: Tỷ lệ sản phẩm hoặc nguyên liệu không thể sử dụng lại, bị loại bỏ.

$$\text{Tỷ lệ phế phẩm} = \frac{\text{Sản phẩm/Nguyên liệu bỏ đi}}{\text{Tổng sản xuất}} \times 100\% \quad (2.4)$$

- *Sigma Level (Mức Sigma)*: Chỉ số đo mức độ biến thiên (độ lệch) của một quy trình so với giới hạn chất lượng cho phép. Càng ít biến thiên, quy trình càng "chuẩn" và độ sigma càng cao.

Các chỉ số này cho phép nhà quản lý giám sát chất lượng theo thời gian thực và ra quyết định cải tiến phù hợp.

2.2. Các công cụ truyền thống trong quản lý chất lượng

Trước khi công nghệ phát triển như trí tuệ nhân tạo hay các mô hình học máy được áp dụng. Các doanh nghiệp đã sử dụng 7 công cụ quản lý chất lượng để phân tích dữ liệu, kiểm soát và cải tiến chất lượng, quy trình trong sản xuất. Đó là những công cụ đơn giản, dễ dàng áp dụng nhưng mang hiệu quả cao trong việc phát hiện, loại bỏ các nguyên nhân gây lỗi

2.2.1 Phiếu kiểm tra (Check Sheet)

Check Sheet									
Model		Part Name				Part No:			
Operator		Location				Model			
S.No	Defect	Date Wise Rejection							Total
		Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday	
1	Defec1	8	8	7	1	5	1	6	36
2	Defec2	5	15	2	7	8	6	2	45
3	Defec3	5	8	5	4	1	9	14	46
4	Defec4	6	3	5	6	1	8	6	35
5	Defec5	6	6	3	3	4	6	8	36
6	Defec6	2	4	9	1	9	2	6	33
Total		32	44	31	22	28	32	42	

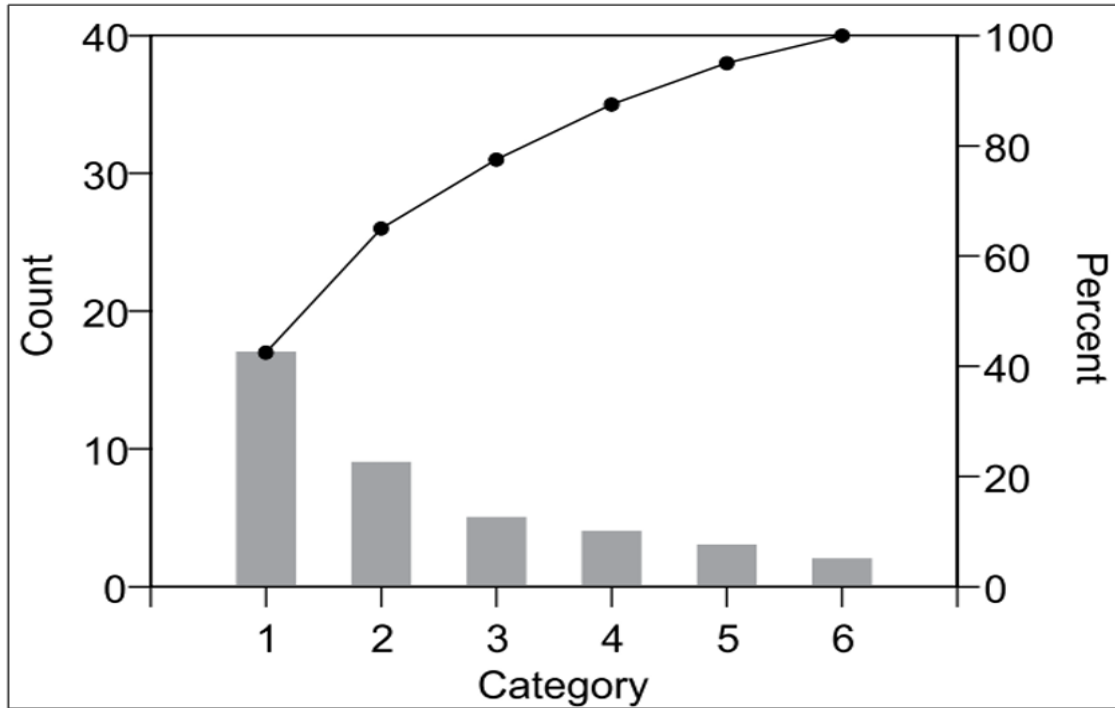
Hình 2.1: Phiếu kiểm tra

(Nguồn: Thu thập từ Internet)

Phiếu kiểm tra là một phương pháp thu thập số liệu, dữ liệu cần thiết để theo dõi các hoạt động đã xảy ra trong quá trình sản xuất và là phương tiện giúp theo dõi xu hướng quá trình một cách khách quan. Theo Neyestani (2017): “Ưu điểm chính của phiếu kiểm tra là rất dễ dàng để áp dụng và hiểu và nó có thể tạo ra một bức tranh rõ ràng về tình huống và tình trạng của tổ chức” [5]. Có nhiều loại phiếu kiểm tra và mỗi loại được sử dụng với từng mục đích và sản phẩm cụ thể. Hiện nay, các loại phiếu kiểm tra thường được sử dụng phổ biến như: bảng kiểm tra phân loại (Classification check sheet), bảng kiểm tra định vị (Defect location check sheet), bảng kiểm tra tần suất (Frequency check sheet), bảng kiểm tra thang đo (Measurement scale check sheet), danh sách kiểm tra (Check list). Trên mỗi phiếu kiểm tra được thiết kế gồm các danh mục vấn đề cần được theo dõi, thông tin và số liệu ghi nhận đơn giản, rõ ràng, nội dung ở dạng con số có ý nghĩa phục vụ cho mục đích kiểm tra. Việc

thu thập dữ liệu trên phiếu kiểm tra rất quan trọng vì đây là bước đầu tiên và cũng là dữ liệu đầu vào phục vụ cho các phương pháp phân tích dữ liệu khác. Đồng thời, trực quan hóa các giá trị giúp phát hiện các vấn đề bất thường trong tổng thể.

2.2.2 Biểu đồ Pareto (Pareto Chart)

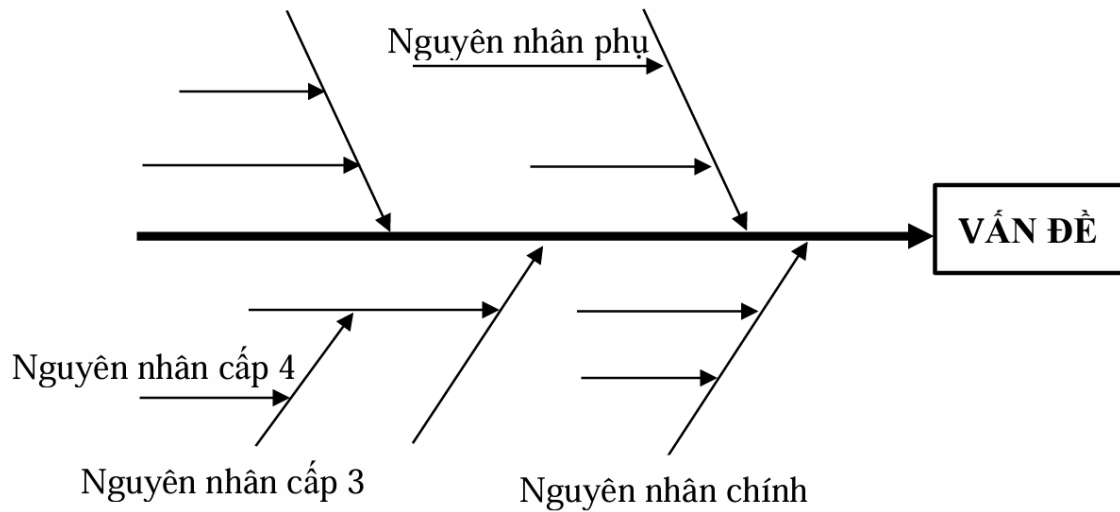


Hình 2.2: Biểu đồ Pareto

(Nguồn: Wilkingson, 2006)

Biểu đồ Pareto, còn gọi là nguyên tắc 80/20, là một công cụ ra quyết định dựa trên quan sát rằng khoảng 80% hậu quả thường bắt nguồn từ 20% nguyên nhân cốt lõi. Ngược lại, trong nhiều trường hợp, chỉ 20% nỗ lực có thể mang lại đến 80% giá trị hoặc hiệu quả của một dự án. Khái niệm này trở nên phổ biến trong lĩnh vực kinh tế và quản lý nhờ công trình của Joseph M. Juran vào năm 1941, lấy cảm hứng từ nghiên cứu của nhà kinh tế học Vilfredo Pareto. Biểu đồ Pareto là đồ thị có dạng hình cột kết hợp với đường cong tích lũy thể hiện vấn đề theo thứ tự từ cao đến thấp giúp xác định các vấn đề hoặc nguyên nhân trọng yếu cần giải quyết. Theo Nicolae và cộng sự (2015), biểu đồ Pareto làm nổi bật các nguyên nhân phổ biến và các lỗi thường xuyên xảy ra nhất từ đó ưu tiên giải quyết chúng góp phần kiểm soát lỗi và nâng cao chất lượng sản phẩm [6].

2.2.3 Biểu đồ nhân quả (Biểu đồ xương cá)

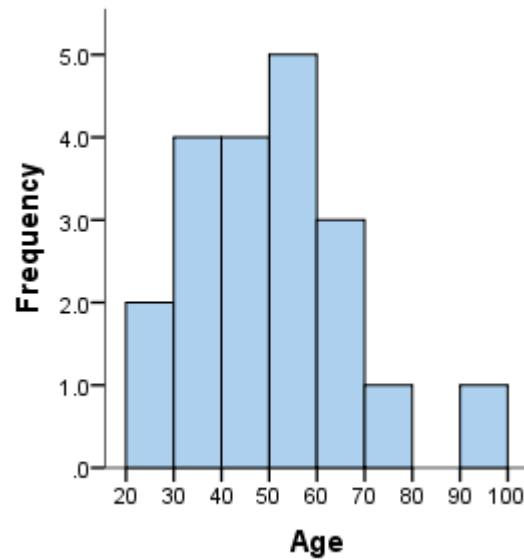


Hình 2.3: Cấu trúc chung của biểu đồ xương cá

(Nguồn: Bilsel và cộng sự, 2012 [7])

Biểu đồ dùng chỉ mối quan hệ nguyên nhân kết quả của một vấn đề. Nó là một công cụ hiệu quả để tạo ra các ý tưởng một cách có hệ thống về nguyên nhân của các vấn đề và trình bày chúng dưới dạng có cấu trúc (Magar, 1014) [8]. Để tìm hiểu nguyên nhân vấn đề thì biểu đồ nhân quả thường xoay quanh các yếu tố như: con người, nguyên vật liệu, phương pháp, máy móc, môi trường. Sử dụng biểu đồ nhân quả để tìm hiểu nguyên nhân từ những yếu tố tổng quan từ đó tìm hiểu sâu đến các yếu tố ảnh hưởng tiềm ẩn của vấn đề.

2.2.4 Biểu đồ mật độ phân bố (Histogram)

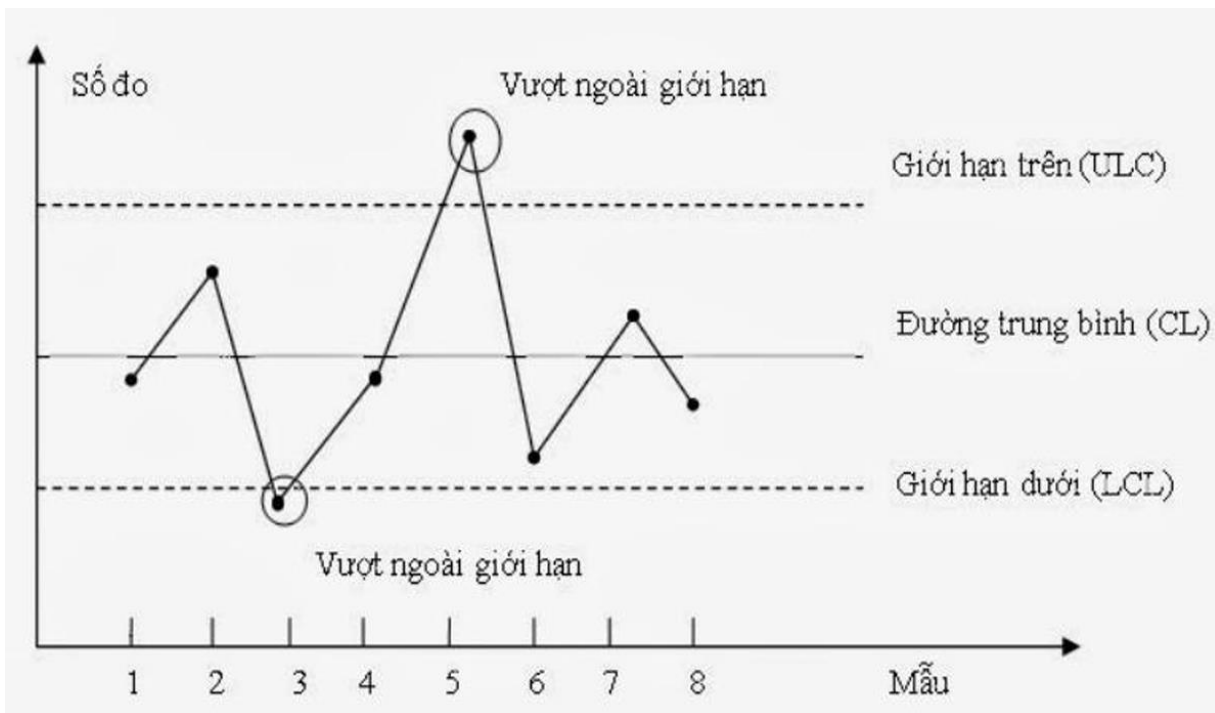


Hình 2.4: Biểu đồ Histogram

(Nguồn: Thu thập từ Internet)

Biểu đồ có dạng cột, thường được sử dụng để biểu diễn phân phối tần suất của một tập dữ liệu liên tục. Thông qua biểu đồ có thể thấy được xu hướng phân phối, biên độ, độ tập trung của dữ liệu và sự phân tán của chúng. Điều này giúp cho doanh nghiệp có thể đánh giá và cải thiện quy trình, chất lượng sản phẩm hiệu quả hơn.

2.2.5 Biểu đồ kiểm soát (Control Chart)



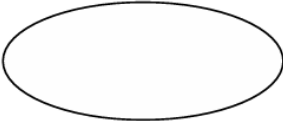
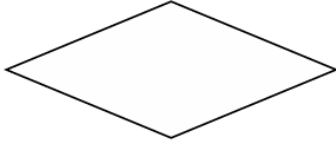
Hình 2.5: Biểu đồ kiểm soát

(Nguồn: Thu thập từ Internet)

Biểu đồ kiểm soát là dạng đồ thị hình gấp khúc thể hiện giá trị trung bình của các đặc tính hoặc khuyết tật mà ta cần kiểm soát. Chúng được sử dụng nhằm theo dõi quá trình biến động nhằm phát hiện vấn đề bất thường trong quá trình sản xuất hoặc đánh giá kết quả thực hiện một dự án nhờ vào sự thay đổi giá trị của các đặc tính trên biểu đồ trong khoảng thời gian nhất định. Cấu trúc biểu đồ kiểm soát bao gồm: đường trung tâm và đường giới hạn kiểm soát được sử dụng để kiểm soát sự biến động của quá trình. Trên các đường này vẽ ra các điểm thể hiện chất lượng hoặc điều kiện quá trình. Nếu các điểm này nằm trong đường giới hạn kiểm soát thì quá trình ổn định và nếu các điểm nằm ngoài thể hiện xu hướng bất thường có nghĩa là đang tồn tại nguyên nhân gây ra lỗi. Theo (Magar, 2014): “Những biến thể này có thể do hai nguyên nhân; Nguyên nhân ngẫu nhiên/cơ hội (không thể ngăn chặn) và nguyên nhân có thể chỉ định (có thể phòng ngừa)” [8].

2.2.6 Lưu đồ (Flowchart)

Bảng 2.1: Ký hiệu cơ bản được quy định trong Flowchart

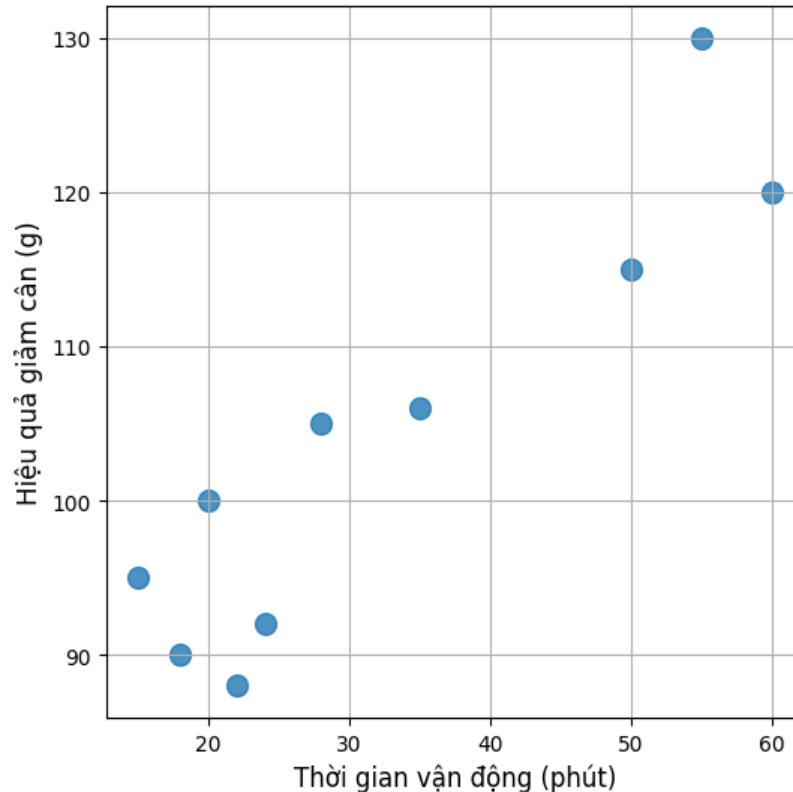
CÁC KÝ HIỆU CƠ BẢN ĐƯỢC QUY ĐỊNH TRONG FLOWCHART		
Tên	Ký hiệu	Ý nghĩa
Hình chữ nhật		Thể hiện các bước trong quy trình sản xuất
Hình oval		Bắt đầu hoặc kết thúc quy trình
Hình thoi		Đưa ra quyết định trong hoạt động
Đường mũi tên		Quan hệ một chiều, sự chỉ dẫn tuần tự giữa các bước trong quy trình

(Nguồn: ANSI, 1985)

Lưu đồ là chuỗi các hoạt động được sắp xếp theo thứ tự tạo thành một quy trình sản xuất hoặc quy trình thực hiện một công việc cụ thể. Theo HACCP (2020): “cần xây dựng một lưu đồ dòng chảy bao gồm tất cả các khâu trong quá trình sản xuất 1 sản phẩm cụ thể, bao gồm cả bất kì công đoạn gia công nào” [9]. Lưu đồ được thiết kế theo các kí hiệu hình dạng chuẩn hóa phù hợp với ý nghĩa của hoạt động tương ứng. Dựa trên dòng chảy lưu đồ mà người đọc có thể dễ dàng nắm bắt được quy trình vì nó thể hiện tiến trình công việc một cách trực quan và đơn giản hóa. Ngoài ra, nó còn giúp nhân viên thực hiện công việc một cách đồng nhất góp phần tạo nên sự ổn định về mặt chất lượng. Trong kiểm soát chất lượng, lưu đồ được ứng dụng để so sánh đối chiếu các công đoạn sản xuất thực tế với quy trình

tiêu chuẩn. Trong quá trình sản xuất có nhiều loại sai lỗi xảy ra, lưu đồ giúp xác định khoanh vùng công đoạn gây ra lỗi nhằm thu hẹp phạm vi tìm hiểu nguyên nhân vấn đề.

2.2.7 Biểu đồ phân tán (Scatter Diagram)



Hình 2.6 : Biểu đồ phân tán

Biểu đồ thể hiện dữ liệu dưới dạng đồ thị. Trong đó, các giá trị của từng biến sẽ biểu diễn thành từng chấm riêng mà không được nối lại với nhau. Do đó, biểu đồ phân tán sẽ thể hiện mối tương quan giữa 2 nhân tố. Từ đó, có thể thấy được sự phụ thuộc của 1 nhân tố vào nhân tố còn lại cũng như mức độ phụ thuộc lẫn nhau.

2.3. Hạn chế của công cụ truyền thống và xu hướng ứng dụng học máy trong kiểm tra chất lượng vải

2.3.1. Hạn chế của các công cụ truyền thống

Mặc dù 7 công cụ quản lý chất lượng truyền thống đóng vai trò quan trọng trong việc cải tiến quy trình và kiểm soát lỗi trong sản xuất. Nhưng chúng vẫn tồn tại nhiều hạn chế, đặc biệt trong bối cảnh sản xuất hiện đại yêu cầu tính chính xác cao, tốc độ nhanh và tự động hóa [10]:

Phụ thuộc vào con người: Các công cụ như phiếu kiểm tra, biểu đồ Pareto,... yêu cầu dữ liệu đầu vào do con người thu thập. Điều này dễ dẫn đến sai sót, thiếu nhất quán hoặc không đầy đủ.

Phụ thuộc vào kỹ năng và kinh nghiệm của người sử dụng: Hiệu quả của các công cụ này phụ thuộc nhiều vào kỹ năng và kinh nghiệm của người sử dụng, dẫn đến khả năng áp dụng không đồng đều. Theo Antony, J., McDermott, O., & Sony, M. (2023) chỉ ra rằng 40% chuyên gia chất lượng thừa nhận đã áp dụng sai các công cụ ngay từ lần đầu tiên, cho thấy sự phụ thuộc vào kỹ năng cá nhân [10].

Thiếu khả năng xử lý dữ liệu hình ảnh: Các công cụ truyền thống không được thiết kế để xử lý dữ liệu phức tạp như ảnh chụp bề mặt vải - loại dữ liệu ngày càng phổ biến trong kiểm tra ngoại quan.

Khó áp dụng trong thời gian thực: Việc tổng hợp và phân tích dữ liệu thủ công mất thời gian, không đáp ứng được yêu cầu kiểm tra tự động trong dây chuyền sản xuất tốc độ cao.

Không phát hiện được lỗi tinh vi hoặc bất thường: Những lỗi nhỏ, khó nhận biết bằng mắt thường hoặc lỗi không tuân theo quy luật cụ thể có thể không bị phát hiện bởi các công cụ QC cơ bản.

2.3.2. Xu hướng ứng dụng học máy trong kiểm tra chất lượng vải

Trước những hạn chế hiện tại, các công nghệ mới như trí tuệ nhân tạo (AI) và đặc biệt là học máy (Machine Learning) đang được nghiên cứu và áp dụng rộng rãi vào quy trình kiểm tra chất lượng trong ngành dệt may, đặc biệt là kiểm tra chất lượng ngoại quan của vải. Việc sử dụng các mô hình học máy không chỉ nâng cao hiệu quả kiểm tra mà còn tối ưu hóa toàn diện quy trình sản xuất và giảm chi phí. Một số xu hướng ứng dụng nổi bật hiện nay bao gồm [11]:

Tự động hóa quy trình kiểm tra: Hệ thống camera kết hợp với mô hình học máy cho phép phát hiện lỗi trên bề mặt vải theo thời gian thực mà không cần sự can thiệp thủ công. Việc tự động hóa quy trình giúp tăng tốc độ kiểm tra, hạn chế sai sót và đảm bảo quá trình sản xuất diễn ra liên tục.

Phân loại lỗi chính xác: Các mô hình học sâu (deep learning), đặc biệt là mạng nơ-ron tích chập (CNN), có khả năng trích xuất đặc trưng từ hình ảnh và phân loại chính xác nhiều lỗi khác nhau như rách vải, sợi thô, tạp chất, hay vết bẩn. Nhờ đó, hệ thống có thể đưa ra đánh giá cụ thể, phù hợp với từng loại lỗi cần xử lý.

Khả năng phát hiện bất thường: Các kỹ thuật như Autoencoder hay One-Class SVM cung cấp khả năng phát hiện các lỗi hiếm gặp, khó nhận biết hoặc không có mẫu lỗi rõ ràng trong dữ liệu huấn luyện. Đây là giải pháp hữu ích trong môi trường sản xuất thực tế, nơi các lỗi thường không xảy ra theo quy luật cố định.

Giảm phụ thuộc vào chuyên gia: Hệ thống kiểm tra dựa trên học máy có khả năng tự học từ dữ liệu, qua đó giảm nhu cầu đào tạo nhân viên có chuyên môn cao và hạn chế rủi ro từ sự đánh giá chủ quan của con người. Điều này góp phần đảm bảo tính nhất quán và cải thiện đáng kể hiệu quả kiểm tra.

Tích hợp vào dây chuyền sản xuất: Các mô hình học máy có thể được triển khai trực tiếp trên hệ thống kiểm tra tự động và dễ dàng tích hợp vào dây chuyền sản xuất công nghiệp. Việc này giúp doanh nghiệp tiết kiệm thời gian, giảm chi phí vận hành, đồng thời tăng cường khả năng kiểm soát chất lượng của sản phẩm đầu ra.

2.3.3 Các loại học máy

2.3.3.1 Học có giám sát (Supervised Learning)

Đối với học có giám sát, mô hình được huấn luyện dựa trên tập dữ liệu có nhãn. Có nghĩa là mỗi mẫu dữ liệu đều sẽ đi kèm với thông tin đầu ra mong muốn.

Mục tiêu của học có giám sát là tìm ra mối quan hệ giữa đầu vào và đầu ra để có thể dự đoán cho dữ liệu mới. Các thuật toán thường được sử dụng trong học có giám sát bao gồm: hồi quy tuyến tính (Linear Regression), hồi quy logistic (Logistic Regression), cây quyết định (Decision Tree), rừng ngẫu nhiên (Random Forest) và máy vector hỗ trợ (Support Vector Machine - SVM).

2.3.3.2 Học không giám sát (Unsupervised Learning)

Học không giám sát là một phương pháp trong học máy, trong đó mô hình tự học từ dữ liệu mà không cần gắn nhãn.

Mục tiêu của phương pháp này là phân tích dữ liệu dựa trên sự tương đồng và khác biệt trong tập dữ liệu đầu vào. Từ đó tìm ra các mẫu, cấu trúc ẩn hoặc mối quan hệ trong dữ liệu mà không cần biết trước kết quả chính xác. Một số thuật toán phổ biến trong học không giám sát là phân cụm K-means, phân tích thành phần chính (PCA - Principal Component Analysis) và phân tích thành phần độc lập (ICA - Independent Component Analysis).

2.4 Cơ sở lý thuyết về mô hình học máy

2.4.1 Giới thiệu về mô hình học máy

Mạng nơ-ron tích chập (CNN) là loại mạng nơ-ron được dùng phổ biến để xử lý hình ảnh. Nó hoạt động bằng cách “quét” qua từng phần nhỏ của ảnh để tìm ra các đặc điểm quan trọng như đường viền, màu sắc hay hoa văn. Mạng CNN gồm các tầng chính như tầng tích chập (phát hiện đặc trưng), tầng gộp (giảm kích thước ảnh) và tầng kết nối đầy đủ (đưa ra kết quả cuối cùng). CNN được ứng dụng rộng rãi trong các bài toán như nhận diện khuôn mặt, phân loại ảnh, phát hiện lỗi sản phẩm và phân đoạn hình ảnh trong y tế.

Mạng nơ-ron hồi quy (RNN) được thiết kế để xử lý các dữ liệu có thứ tự, ví dụ như văn bản, âm thanh hoặc chuỗi thời gian. RNN có khả năng “ghi nhớ” thông tin trước đó để hiểu được ngữ cảnh hiện tại. Tuy nhiên, RNN truyền thống có thể bị quên khi chuỗi quá dài, vì vậy người ta phát triển các biến thể như LSTM và GRU để cải thiện khả năng ghi nhớ. Mạng RNN thường được dùng trong dịch ngôn ngữ, nhận diện giọng nói và dự báo dữ liệu theo thời gian (như giá cổ phiếu, lượng tiêu thụ điện...).

Mạng Transformer là một kiến trúc hiện đại hơn, xử lý dữ liệu dựa trên cơ chế “tự chú ý” (self-attention), tức là nó biết phần nào trong dữ liệu là quan trọng hơn để tập trung vào. Không giống như RNN, Transformer xử lý toàn bộ chuỗi cùng lúc, giúp nhanh hơn và hiệu quả hơn. Transformer rất mạnh trong các bài toán liên quan đến ngôn ngữ như dịch văn bản, tạo văn bản tự động và là nền tảng cho các mô hình lớn như GPT hay BERT.

Mạng tự mã hóa (Autoencoder) là một mô hình học cách “nén” dữ liệu lại rồi “giải nén” ra giống như ban đầu. Mục tiêu là giúp máy học cách hiểu và giữ lại những đặc điểm quan trọng nhất của dữ liệu. Autoencoder được dùng để giảm kích thước dữ liệu, phát hiện dữ liệu bất thường hoặc tái tạo lại ảnh bị nhiễu. Trong sản xuất, autoencoder có thể giúp phát hiện các sản phẩm lỗi khi ảnh tái tạo khác ảnh ban đầu.

Mạng GAN (Generative Adversarial Network) là một mạng đặc biệt gồm 2 phần: một phần tạo dữ liệu giả (Generator) và một phần cố gắng phát hiện đâu là dữ liệu thật hay giả (Discriminator). Hai phần này “thi đấu” với nhau, làm cho dữ liệu giả ngày càng giống thật. GAN được dùng để tạo ảnh, chỉnh sửa ảnh, tạo khuôn mặt giả và tạo ra dữ liệu mới khi thiếu dữ liệu thực tế.

2.4.2 Các khái niệm cốt lõi hỗ trợ việc xây dựng và huấn luyện mô hình

Hàm mất mát (Loss Function): Một công cụ quan trọng dùng để đo lường mức độ sai lệch giữa kết quả dự đoán của mô hình và giá trị thực tế. Mục tiêu của quá trình huấn luyện mô hình là làm cho sai lệch này càng nhỏ càng tốt. Với bài toán hồi quy - tức là dự đoán một giá trị liên tục - một trong những hàm mất mát phổ biến nhất là Mean Squared Error (MSE), tính trung bình bình phương sai số giữa các dự đoán và giá trị thực. Trong các bài toán phân loại, chẳng hạn như nhận diện chữ viết hoặc phân biệt email spam, người ta thường sử dụng Cross-Entropy Loss để đo lường hiệu quả. Trong các mô hình học sâu phức tạp hơn, như Variational Autoencoders (VAE), có thể sử dụng hàm mất mát chuyên biệt như Kullback-Leibler Divergence, cho phép mô hình học được phân phối xác suất tiềm ẩn trong dữ liệu.

Tối ưu hóa (Optimization): Quá trình điều chỉnh các tham số của mô hình (như trọng số trong mạng nơ-ron) nhằm mục tiêu giảm thiểu giá trị của hàm mất mát. Phương pháp cơ bản nhất là Gradient Descent, trong đó mô hình cập nhật các tham số theo hướng giảm dần của độ dốc hàm mất mát. Phiên bản cải tiến là Stochastic Gradient Descent (SGD), giúp tăng tốc quá trình học bằng cách sử dụng các mẫu nhỏ (mini-batch) thay vì toàn bộ dữ liệu. Ngoài ra, còn có các thuật toán tối ưu hóa tiên tiến hơn như Adam hoặc RMSprop, rất hiệu quả trong việc huấn luyện các mô hình học sâu nhờ khả năng điều chỉnh tốc độ học một cách thông minh. Tuy nhiên, việc tối ưu hóa trong học sâu thường phức tạp do có hàng triệu tham số và dễ gặp hiện tượng như gradient biến mất (vanishing gradient).

Lan truyền ngược (Backpropagation): Phương pháp cốt lõi trong việc huấn luyện các mô hình học sâu. Nó cho phép tính toán hiệu quả gradient của hàm mất mát đối với từng tham số trong mô hình. Quá trình này diễn ra từ đầu ra quay ngược về đầu vào, điều chỉnh từng tầng một trong mạng nơ-ron. Nhờ vào backpropagation, các mô hình có thể tự học cách điều chỉnh để cải thiện kết quả dự đoán. Phương pháp này không chỉ được dùng trong các mạng nơ-ron đơn giản (trong học máy) mà còn là nền tảng cho mạng nơ-ron sâu và các mô hình học sâu phức tạp.

Overfitting và Underfitting: Hai vấn đề phổ biến trong huấn luyện mô hình là overfitting và underfitting. Overfitting xảy ra khi mô hình học quá kỹ các chi tiết trong dữ liệu huấn luyện, dẫn đến việc không hoạt động tốt trên dữ liệu mới - nói cách khác, mô hình "học vẹt". Trong khi đó, underfitting là khi mô hình học chưa đủ, không thể nắm bắt các quy luật trong dữ liệu. Để giải quyết hai vấn đề này, các kỹ thuật như regularization (chẳng hạn L1, L2 hoặc Dropout trong học sâu) được sử dụng để tránh quá khớp. Ngoài ra, các biện pháp

như cross-validation và việc thu thập thêm dữ liệu chất lượng cũng giúp cải thiện mô hình. Trong học sâu, những kỹ thuật như Batch Normalization và Weight Decay còn được thêm vào để tăng độ ổn định khi huấn luyện mô hình lớn.

Hàm kích hoạt (Activation Functions): Hàm kích hoạt giúp mạng nơ-ron có khả năng học được các mối quan hệ phi tuyến tính trong dữ liệu - đây là yếu tố quan trọng để giải quyết các vấn đề phức tạp. Một số hàm kích hoạt thường gặp gồm ReLU (Rectified Linear Unit), Sigmoid, Tanh và Softmax. Trong đó, ReLU được ưa chuộng nhất trong học sâu vì nó giảm thiểu hiện tượng gradient biến mất, giúp mô hình học nhanh và hiệu quả hơn. Với các bài toán phân loại nhiều lớp, hàm Softmax thường được dùng ở tầng đầu ra để tạo ra xác suất cho từng lớp.

Kỹ thuật cải thiện trong học sâu:

Batch Normalization: Chuẩn hóa đầu ra của các tầng để ổn định huấn luyện.

Dropout: Ngẫu nhiên bỏ qua một số nơ-ron trong huấn luyện để ngăn overfitting.

2.4.3 Khái niệm chỉ số đánh giá

SSIM (Structural Similarity Index): chỉ số được sử dụng để đo lường mức độ tương đồng giữa hai hình ảnh. SSIM được thiết kế nhằm mô phỏng cách con người cảm nhận sự khác biệt giữa các hình ảnh, không chỉ dựa vào sai số điểm ảnh đơn lẻ (như MSE hay PSNR), mà còn dựa vào các cấu trúc, độ sáng và độ tương phản cục bộ [12].

Công thức tổng quát của SSIM giữa hai ảnh x và y :

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (2.5)$$

μ_x, μ_y : độ sáng trung bình của ảnh x và y

σ_x^2, σ_y^2 : phương sai

σ_{xy} : hiệp phương sai

C_1, C_2 : các hằng số ổn định hóa

Max Difference (sai số tuyệt đối lớn nhất): là chỉ số đánh giá độ lệch giữa hai tập dữ liệu, phản ánh mức sai khác cực đại tại bất kỳ điểm nào trong toàn bộ không gian dữ liệu. Thay vì đo trung bình hoặc mức độ tương quan tổng thể, Max Difference tập trung vào trường hợp có sai số lớn nhất, cho thấy điểm có mức chênh lệch nghiêm trọng nhất giữa dữ liệu

thực và dữ liệu dự đoán (giữa ảnh gốc và ảnh xử lý). Chỉ số này thường được sử dụng trong các tình huống yêu cầu giám sát chất lượng nghiêm ngặt hoặc đảm bảo tính an toàn kỹ thuật, chẳng hạn như trong thị giác máy công nghiệp, xử lý ảnh y tế, hoặc kiểm thử hệ thống số [13].

Một điểm mạnh nổi bật của Max Difference là khả năng phát hiện các lỗi cục bộ mà các chỉ số đánh giá trung bình dễ bỏ qua, giúp nhận diện hiệu quả các trường hợp ngoại lệ hoặc sai sót vượt ngưỡng cho phép. Tuy nhiên, do chỉ phản ánh một điểm duy nhất, nó không cung cấp cái nhìn toàn diện về chất lượng hoặc mức độ khái quát của sai số trên toàn bộ dữ liệu. Công thức của Max difference:

$$\text{Max Difference} = \max_i |x_i - \hat{x}_i| \quad (2.6)$$

x_i : giá trị thực tại điểm i

$\hat{x}_i$: giá trị dự đoán/bị biến đổi tại điểm i

Bảng 2.2: So sánh tính năng giữa chỉ số SSIM và Max Difference

Tiêu chí	SSIM (Structural Similarity Index)	Max Difference (Maximum Absolute Error)
Đặc trưng đo lường	Tương đồng theo cấu trúc, độ sáng, tương phản	Sai số điểm cực đại (worst-case error)
Phạm vi giá trị	Từ 0 đến 1 (gần 1 là giống nhau hơn)	Lớn hơn hoặc bằng 0 (càng lớn sai số càng lớn)
Nhạy với lỗi cục bộ	Tính trung bình theo vùng	Chỉ một điểm sai lệch cũng làm tăng chỉ số
Khả năng phản ánh cảm nhận thị giác	Tô phỏng nhận thức thị giác của con người	Chỉ đơn thuần đo mức sai số lớn nhất
Tính chất toàn cục	Đánh giá ảnh tổng thể dựa trên vùng	Tập trung vào một điểm sai lệch lớn nhất
Khả năng giải thích mô hình	Khó định vị sai lệch cụ thể	Dễ xác định vị trí có sai lệch lớn nhất
Ưu điểm	Phản ánh tốt chất lượng ảnh theo cảm nhận con người	Đơn giản, rõ ràng, dễ phát hiện sai số cực trị
Nhược điểm	Bỏ qua sai số lớn tại một vài điểm nếu tổng thể vẫn tốt	Không đánh giá được chất lượng tổng thể hay độ tương đồng tổng thể

2.5 Giới thiệu về mô hình

2.5.1 Mô hình được sử dụng

Mô hình Autoencoder (AE) - một kiến trúc học sâu không giám sát được thiết kế để học biểu diễn nén của dữ liệu đầu vào và tái tạo lại đầu ra tương tự với dữ liệu gốc. Mô hình bao gồm hai thành phần chính: Encoder (mã hóa) và Decoder (giải mã) với các lớp tích chập (convolution) và chuyển vị tích chập (transpose convolution) được xếp chồng nhằm giảm dần kích thước đặc trưng và sau đó khôi phục lại ảnh ban đầu.

2.5.2 Lí do chọn mô hình

Phù hợp với bài toán không giám sát: Trong nhiều bài toán công nghiệp thực tế, dữ liệu lỗi rất khó thu thập hoặc không được gán nhãn đầy đủ. Việc xây dựng một tập dữ liệu lỗi hoàn chỉnh là không khả thi do chi phí cao và tính đa dạng của các loại lỗi. Vì vậy, việc sử dụng một mô hình không giám sát, cụ thể là Autoencoder, giúp giải quyết bài toán này một cách hiệu quả. Mô hình chỉ cần học trên dữ liệu "tốt" và sử dụng sai lệch tái tạo để phát hiện bất thường, hoàn toàn không phụ thuộc vào dữ liệu lỗi trong quá trình huấn luyện [14].

Khả năng học biểu diễn đặc trưng mạnh mẽ: Autoencoder có thể học được các đặc trưng quan trọng nhất của ảnh đầu vào như biên, hình dạng, hoa văn và kết cấu thông qua quá trình nén và giải nén. Nhờ vào các lớp tích chập trong Encoder, mô hình không chỉ giảm được chiều dữ liệu mà còn trích xuất hiệu quả các thông tin có ý nghĩa. Việc này giúp mô hình trở nên nhạy bén với những điểm khác biệt nhỏ trên bề mặt sản phẩm - những chi tiết mà các thuật toán truyền thống rất khó nhận diện [14].

Khả năng tái tạo ảnh chính xác cao: Với kiến trúc mạng nơ-ron tích chập (CNN), Autoencoder có khả năng tái tạo lại ảnh đầu vào với độ chính xác cao. Việc kết hợp nhiều lớp Conv2D và ConvTranspose2D giúp mô hình học và bảo toàn được cả đặc điểm hình học lẫn kết cấu của ảnh gốc. Điều này đặc biệt cần thiết trong các ứng dụng kiểm tra lỗi bề mặt, nơi các sai lệch dù rất nhỏ cũng có thể ảnh hưởng đến chất lượng sản phẩm. Khả năng tái tạo chính xác của mô hình là nền tảng quan trọng để so sánh và phát hiện các dị thường.

Hàm mất mát chuyên biệt cho bài toán phát hiện lỗi: Một điểm nổi bật của mô hình là việc xây dựng hàm mất mát FabricReconstructionLoss, kết hợp giữa nhiều thành phần như MS-SSIM, Edge loss, Texture loss và Pixel loss. MS-SSIM giúp mô hình đánh giá độ tương đồng cấu trúc một cách giống với cách con người nhìn nhận hình ảnh. Trong khi đó, Sobel edge và Gaussian blur giúp mô hình nhạy với sự thay đổi ở biên và kết cấu - những yếu tố thường là vị trí xuất hiện lỗi. Việc kết hợp các thành phần này làm tăng khả năng phát hiện dị thường, giảm thiểu tỷ lệ bỏ sót lỗi (false negative).

Tốc độ huấn luyện và triển khai nhanh: So với các mô hình phức tạp như GAN hoặc Transformer, Autoencoder có cấu trúc đơn giản hơn, ít tham số hơn và dễ huấn luyện hơn. Điều này rút ngắn đáng kể thời gian huấn luyện mô hình mà vẫn đảm bảo được hiệu quả. Ngoài ra, khi triển khai thực tế trong dây chuyền sản xuất, yêu cầu tốc độ xử lý thời gian thực là rất cao. Mô hình Autoencoder đáp ứng tốt yêu cầu này nhờ khả năng suy luận nhanh và độ ổn định cao trong môi trường công nghiệp.

Khả năng mở rộng và điều chỉnh linh hoạt: Mô hình Autoencoder có kiến trúc mở, cho phép điều chỉnh dễ dàng các yếu tố như số lượng tầng, kích thước đầu vào, loại hàm mất mát hay hàm kích hoạt. Điều này giúp mô hình thích ứng tốt với nhiều loại dữ liệu khác nhau ngoài hình ảnh vải, như kim loại, nhựa, hoặc các sản phẩm công nghiệp khác. Nhờ tính linh hoạt này, mô hình có thể được mở rộng và tái sử dụng cho nhiều bài toán kiểm tra lỗi khác nhau mà không cần xây dựng lại từ đầu.

2.5.3 Tính ưu việt của mô hình

Không yêu cầu dữ liệu lỗi trong quá trình huấn luyện: Một trong những ưu điểm lớn nhất của mô hình Autoencoder là khả năng huấn luyện hoàn toàn trên dữ liệu tốt mà không cần dữ liệu lỗi. Trong thực tế, việc thu thập dữ liệu có lỗi là rất tốn kém, mất thời gian và đôi khi không thể thực hiện đầy đủ do lỗi xảy ra hiếm hoặc không lường trước được. Mô hình Autoencoder học cách nén và tái tạo ảnh "bình thường", từ đó phát hiện ra các vùng bất thường nếu ảnh không thể được tái tạo chính xác. Điều này giúp tiết kiệm chi phí và thời gian trong quá trình xây dựng hệ thống kiểm tra lỗi [14].

Khả năng tái tạo ảnh đầu vào với độ chính xác cao: Với cấu trúc mạng tích chập nhiều tầng, mô hình có khả năng tái tạo ảnh đầu vào với độ trung thực cao, giữ lại được chi tiết quan trọng về biên, hình dạng và kết cấu. Khả năng tái tạo tốt giúp mô hình dễ dàng phát hiện ra các sai khác nhỏ giữa ảnh gốc và ảnh tái tạo - đây chính là cơ sở để phát hiện lỗi trong các vùng bị hư hỏng, rách, hay sai lệch kết cấu. Đây là yếu tố then chốt để ứng dụng trong kiểm tra bề mặt công nghiệp, nơi những sai sót nhỏ nhất cũng cần được phát hiện chính xác.

Nhạy bén với các sai lệch nhỏ nhờ kết hợp nhiều loại mất mát: Hàm mất mát được thiết kế riêng biệt cho bài toán phát hiện lỗi là một điểm nâng cao đáng kể. Việc kết hợp giữa nhiều thành phần như MS-SSIM, edge loss và texture loss giúp mô hình không chỉ đánh giá điểm ảnh mà còn hiểu sâu hơn về cấu trúc hình ảnh. MS-SSIM giúp mô hình nhận diện sự thay đổi cấu trúc tổng thể, edge loss tập trung vào các đường biên - nơi dễ phát sinh lỗi, trong khi texture loss giúp phát hiện thay đổi nhỏ trong hoa văn hoặc chất liệu. Nhờ đó, mô hình nhạy bén hơn với các sai lệch mà mắt thường đôi khi không nhận ra được.

Tốc độ xử lý nhanh, phù hợp triển khai thực tế: So với các mô hình học sâu khác như GAN hay Transformer, Autoencoder có số lượng tham số vừa phải, cấu trúc đơn giản và dễ triển khai. Trong môi trường sản xuất công nghiệp, nơi yêu cầu kiểm tra hàng trăm đến hàng nghìn sản phẩm mỗi giờ, yếu tố thời gian là rất quan trọng. Mô hình Autoencoder có thể

chạy nhanh trên GPU hoặc thậm chí CPU hiện đại mà vẫn đảm bảo độ chính xác cao. Điều này giúp đảm bảo hiệu quả hoạt động trong dây chuyền kiểm tra lỗi thực tế mà không cần phần cứng quá phức tạp.

Khả năng tổng quát hóa tốt cho dữ liệu mới: Không học ghi nhớ hình ảnh cụ thể, mà học cách nắm bắt "bản chất" của các đặc trưng ảnh không lỗi. Nhờ đó, khi gặp ảnh mới chưa từng xuất hiện trong quá trình huấn luyện, mô hình vẫn có thể tái tạo chính xác nếu đó là ảnh không lỗi, hoặc tạo ra sai lệch rõ ràng nếu đó là ảnh có lỗi. Khả năng tổng quát hóa này giúp mô hình hoạt động ổn định và tin cậy với dữ liệu mới trong quá trình triển khai.

Dễ dàng điều chỉnh, tùy biến và mở rộng: Autoencoder là một kiến trúc có tính linh hoạt cao. Tùy theo yêu cầu cụ thể của từng loại sản phẩm hay chất liệu, kiến trúc mạng có thể được mở rộng (tăng số tầng, thay đổi bộ lọc), thay đổi kích thước ảnh đầu vào, hoặc điều chỉnh hàm mất mát để phù hợp hơn với mục tiêu kiểm tra. Điều này giúp mô hình dễ dàng mở rộng và tái sử dụng cho nhiều bài toán công nghiệp khác nhau như phát hiện lỗi trên kim loại, nhựa, giấy hoặc các bề mặt có hoa văn phức tạp.

2.6 Nghiên cứu liên quan

Theo Bergmann và cộng sự (2018), “Cải thiện phân đoạn lỗi không giám sát bằng cách áp dụng tính tương đồng về cấu trúc cho bộ mã hóa tự động” [15].

Vấn đề: Autoencoder (AE) là phương pháp phổ biến trong phân đoạn lỗi không giám sát, thường sử dụng sai số tái tạo từng pixel. Tuy nhiên, cách tiếp cận này dễ tạo sai số lớn tại các vùng cạnh bị lệch nhẹ và bỏ sót các lỗi có thay đổi cấu trúc nhưng giữ nguyên cường độ điểm ảnh. Điều này làm giảm hiệu quả trong các tình huống thực tế.

Phương pháp: Sử dụng hàm mất mát dựa trên chỉ số SSIM, giúp phát hiện tốt hơn các lỗi thông qua việc so sánh độ sáng, tương phản và cấu trúc hình ảnh. Phương pháp này cho thấy hiệu quả vượt trội trên các bộ dữ liệu vải dệt và vật liệu nano.

SSIM là một phép đo khoảng cách được thiết kế để nắm bắt được độ tương đồng về mặt nhận thức, ít nhạy cảm hơn với sự căn chỉnh cạnh và coi trọng sự khác biệt nổi bật giữa đầu vào và quá trình tái tạo. Nó nắm bắt được sự phụ thuộc lẫn nhau giữa các vùng pixel cục bộ, hoàn toàn bị bỏ qua bởi các phương pháp phân đoạn lỗi không giám sát hiện đại dựa trên bộ mã hóa tự động với tổn thất trên mỗi pixel. Chúng tôi đánh giá mức tăng hiệu suất thu được bằng cách sử dụng SSIM làm hàm mất mát trên hai tập dữ liệu kiểm tra thực tế có

liên quan đến công nghiệp và chứng minh mức tăng hiệu suất đáng kể so với các phương pháp từng pixel.

SSIM ít nhạy cảm với sai số nhỏ về vị trí cạnh vì so sánh các vùng cục bộ và kết hợp ba yếu tố: độ sáng, độ tương phản và cấu trúc. Việc sử dụng SSIM thay cho hàm mất mát trên từng pixel cải thiện đáng kể hiệu quả trên bộ dữ liệu nanofibrous thực tế và bộ dữ liệu vải dệt công khai. SSIM giúp nâng cao chất lượng phân đoạn từ gần như không sử dụng được lên ngang bằng với các phương pháp tiên tiến khác, ngay cả trong phân đoạn lỗi không giám sát, dựa trên kiến thức tiên nghiệm về hình ảnh tự nhiên.

Thực nghiệm triển khai: Mỗi mô hình autoencoder được huấn luyện trên 10.000 ảnh về vật liệu sợi nano (NanoTWICE) và một bộ dữ liệu mới về vải dệt không lỗi (patches 128×128), sử dụng thuật toán ADAM trong 200 epoch với learning rate 2×10^{-4} . Mô hình được so sánh gồm autoencoder dùng hàm mất mát L2, autoencoder có điều kiện SSIM, variational autoencoder (VAE) và feature matching autoencoder. Quá trình đánh giá được thực hiện bằng cách tái tạo ảnh kiểm thử, tạo bản đồ sai số (residual map) và tính đường cong ROC cùng giá trị AUC.

Kết quả: Sử dụng SSIM làm số liệu đánh giá và mất mát vượt trội hơn đáng kể so với tất cả các kiến trúc đã thử nghiệm khác. Chỉ cần thay đổi hàm mất mát, AUC đạt được cải thiện từ 0,688 lên 0,966 trên tập dữ liệu vật liệu nano sợi.

Pongetti, J., Kipouros, T., Emmanuelli, M., Ahlfeld, R., & Shahpar, S. (2021, tháng 6). Sử dụng bộ mã hóa tự động và hợp nhất đầu ra để cải thiện các mô hình học máy cho các ứng dụng máy móc tuabin. Trong Turbo Expo: Power for Land, Sea, and Air (Tập 84935, trang V02DT36A018). Hiệp hội Kỹ sư Cơ khí Hoa Kỳ [16].

Vấn đề: Trong lĩnh vực thiết kế máy móc tuabin, việc sử dụng các mô phỏng động lực học chất lưu (CFD) là cần thiết nhưng vô cùng tốn kém và thời gian. Khi áp dụng học máy để hỗ trợ hoặc thay thế một phần của quá trình này, độ chính xác của mô hình học máy trở thành yếu tố then chốt, đặc biệt là khi dữ liệu đào tạo bị hạn chế. Nghiên cứu đặt ra mục tiêu cải thiện độ chính xác của các mô hình học máy trong việc dự đoán các đại lượng hiệu suất (QoI) - ví dụ như hiệu suất đoạn nhiệt - cho các cánh máy nén siêu âm có tải trọng cao.

Phương pháp: Phương pháp đề xuất kết hợp hai kỹ thuật: autoencoder và hợp nhất đầu ra. Autoencoder được sử dụng để mã hóa hình học 3D của cánh quạt thành các tham số tiềm

ân, cho phép biểu diễn hình dạng phức tạp hiệu quả hơn so với các tham số thiết kế truyền thống (EDP). Các tham số này giúp cải thiện khả năng dự đoán của mô hình học máy.

Tiếp theo, hợp nhất đầu ra được áp dụng bằng cách xây dựng hai mô hình học máy liên tiếp. Mô hình chính dự đoán các trường vật lý trung gian (áp suất, nhiệt độ), trong khi mô hình phụ sử dụng các đầu ra này cùng với thông tin hình học để dự đoán đại lượng quan tâm (QoI). Cách tiếp cận này cho phép tích hợp thông tin vật lý sâu hơn vào mô hình dự đoán.

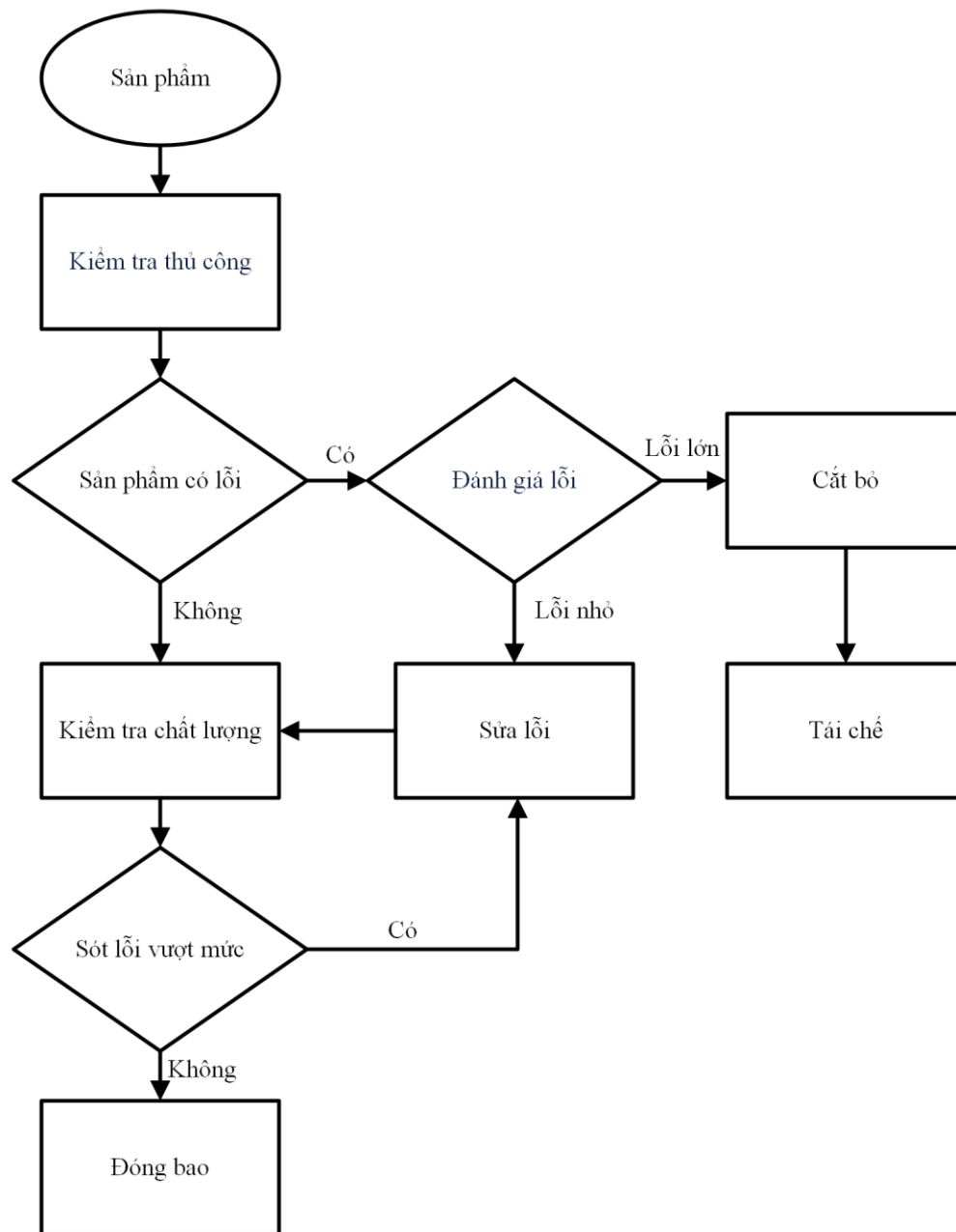
Để tăng hiệu quả, các biến đầu vào được chọn lọc dựa trên mức độ tương quan với QoI, từ đó giảm nhiễu và nâng cao độ chính xác. Phương pháp này giúp cải thiện độ chính xác mô hình trong bối cảnh dữ liệu đào tạo hạn chế.

Thực nghiệm triển khai: Dữ liệu được xây dựng từ 390 mô phỏng CFD của cánh quạt NASA Rotor 37. Các bộ autoencoder được huấn luyện trên 4000 lưới bề mặt được tạo bằng PADRAM. Các mô hình học máy được triển khai bằng phần mềm Monolith, sử dụng các framework như TensorFlow cho mạng nơ-ron và Sklearn cho hồi quy Gaussian (GPR). Các mô hình cơ sở và mô hình có tích hợp autoencoder và hợp nhất đầu ra đều được so sánh dựa trên độ chính xác dự đoán QoI.

Kết quả: Các mô hình có sử dụng autoencoder và hợp nhất đầu ra đã vượt trội rõ rệt so với các mô hình cơ sở. Cụ thể, sai số MSE giảm đến 67% đối với các giá trị QoI cao, hệ số Pearson và R^2 cũng tăng từ 5 - 12% và hiệu suất tổng thể tương đương với mô hình cơ sở dù sử dụng ít hơn 14% số mô phỏng CFD. Thiết kế tối ưu hóa cuối cùng đạt được QoI cao hơn đáng kể so với ban đầu, lên đến 86.799% (so với 84.406%).

CHƯƠNG 3: VẤN ĐỀ DOANH NGHIỆP

3.1 Hiện trạng xử lý lỗi vải ngoại quan



Hình 3.1: Sơ đồ hiện trạng kiểm tra chất lượng ngoại quan tại XDD Textile

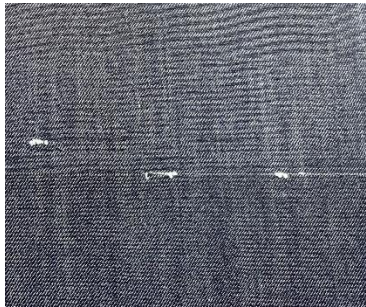
Hiện nay, phương pháp kiểm tra chất lượng ngoại quan tại xưởng thành phẩm của công ty vẫn chủ yếu được thực hiện theo hình thức thủ công, tức là công nhân trực tiếp quan sát bề mặt vải bằng mắt thường trong quá trình cuộn vải di chuyển trên máy kiểm. Công nhân có nhiệm vụ phát hiện và ghi nhận các lỗi phổ biến như: tạp chất, sợi thô, đứt sợi, kết đầu, vết bẩn, lỗ thủng...

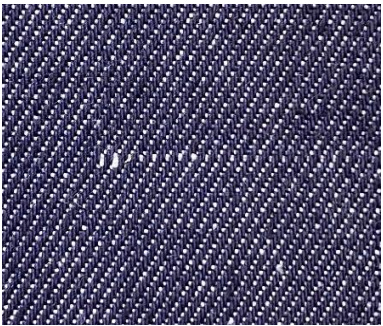
Tuy nhiên, phương pháp kiểm tra thủ công này tồn tại nhiều hạn chế đáng kể:

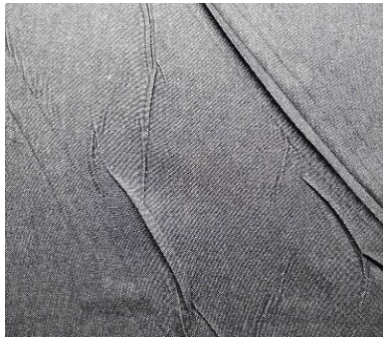
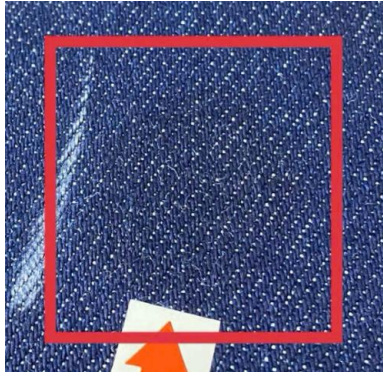
- Phụ thuộc vào con người: kinh nghiệm, khả năng tập trung và tình trạng sức khỏe của người kiểm tra không giống nhau, khiến kết quả kiểm tra mang tính chủ quan và thiếu tính ổn định.
- Khả năng bỏ sót lỗi cao: đối với những lỗi nhỏ, mờ hoặc có hình dạng phức tạp mà mắt thường khó phát hiện trong điều kiện làm việc kéo dài.
- Hiệu quả chưa tối ưu: Nếu tỷ lệ lỗi lớn hoặc số lượng sản phẩm cần sửa lỗi cao, sẽ dẫn đến lãng phí thời gian và chi phí, đồng thời làm chậm tiến độ xử lý. Quy trình còn lặp lại ở bước “Sót lỗi vượt mức”, cho thấy khả năng tồn tại lỗi ngay cả sau nhiều lần kiểm tra. Bên cạnh đó, tốc độ kiểm tra bị giới hạn do phụ thuộc vào thao tác thủ công, đòi hỏi nhiều nhân sự luân phiên làm việc, từ đó làm tăng chi phí vận hành và ảnh hưởng tiêu cực đến năng suất chung của dây chuyền sản xuất.

3.2 Lỗi thông thường

Bảng 3.1: Các lỗi thường gặp

STT	Các lỗi thường gặp	Hình ảnh	Nguyên nhân
1	Đứt sợi		Có bông phé, vệ sinh máy không đúng chỗ dẫn đến đứt sợi
2	Nhảy sợi		Lực căng sợi ngang/quần không đều
3	Kết đầu		Mối nối không được xử lý tốt
4	Sợi dọc thô		Sợi có độ mảnh không đều/ chưa được hồ

5	Sợi ngang co rút		Lực căng của sợi ngang quá chặt. Lực căng sợi phé nhỏ, xuyên sai sợi phé
6	Mật độ sợi ngang thưa		Sợi ngang bị căng Sợi ngang đứt Vệ sinh máy không đúng chỗ
7	Tạp chất		Trong quá trình dệt có bông bay vào
8	Vết mài		Trong quá trình sản xuất/vận chuyển, vải bị va chạm

9	Lỗ thủng		Kim, trục lăn hoặc dao cắt có thể vô tình làm rách vải trong quá trình xử lý
10	Vết gãy		Lỗi dệt hoặc lực tác động mạnh trong quá trình vận chuyển
11	Vết dầu		Dầu bám trên bề mặt lô ép, truyền lên mặt vải khi lăn qua

3.3 Dữ liệu thống kê tỉ lệ lỗi

Bảng 3.2: Dữ liệu thống kê tỉ lệ lỗi

Các lỗi	Tháng 11	Tháng 12	Tháng 1	Tháng 2	Tháng 3	Tháng 4	Tổng
Đứt sợi	800	799	1096	718	998	468	4879
Nhảy sợi	930	672	720	529	695	717	4263
Kết đầu	856	768	820	826	912	317	4499
Sợi dọc thô	830	811	823	711	995	982	5152
Sợi ngang co rút	1154	1253	1824	1773	1929	2067	10000
Mật độ sợi ngang thừa	1307	988	1220	1063	1088	1290	6956
Tạp chất	880	1920	828	1050	812	593	6083
Vết mài	1923	685	2125	1847	1932	1397	9909
Lỗi gợn sóng	1134	1456	1890	1727	1200	1954	9361
Vết hồ	567	884	620	978	996	879	4924
Vết dầu	865	905	702	956	896	679	5003
Tổng lỗi	11246	11141	12668	12178	12453	11343	71029

(Nguồn: Thu thập từ Công ty XDD Textile)

Dựa trên số liệu thống kê lỗi trong 6 tháng (từ tháng 11 đến tháng 4), có thể thấy tổng số lỗi được ghi nhận là 71.029 lỗi, tương đương trung bình khoảng 11.838 lỗi mỗi tháng. Điều này cho thấy hiện tượng lỗi trong quá trình sản xuất vẫn đang tồn tại với tần suất đáng kể. Tháng 1 và tháng 2 là thời điểm ghi nhận số lỗi cao nhất, với 12.668 lỗi và 12.178 lỗi tương ứng.

Từ số liệu trên, có thể rút ra rằng:

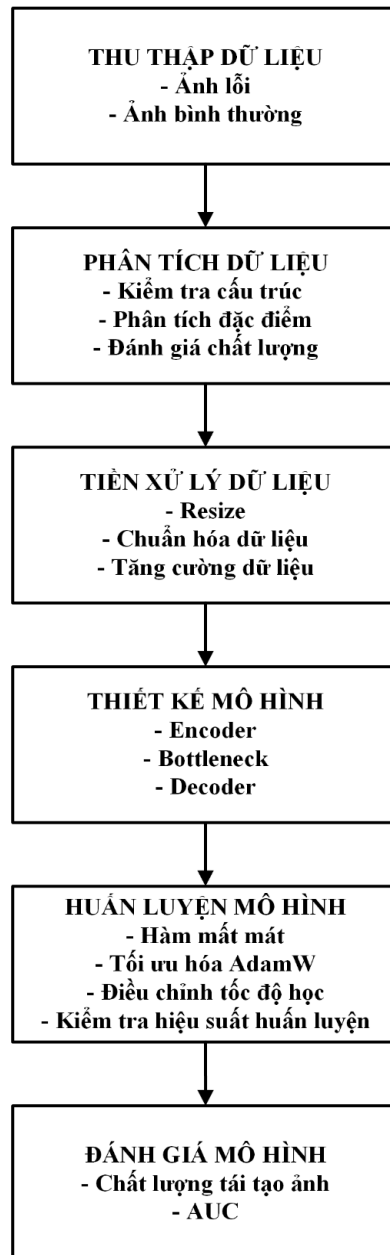
- Việc phát sinh lỗi trong sản xuất là một thực trạng hiện hữu,
- Một số loại lỗi có xu hướng lặp lại và chiếm tỷ lệ lớn,
- Công tác giám sát và kiểm tra lỗi có thể đang đối mặt với áp lực lớn về khối lượng.

Điều này cho thấy tiềm năng và nhu cầu thực tế trong việc ứng dụng các phương pháp tự động hóa, chẳng hạn như thị giác máy tính và học sâu, để hỗ trợ phát hiện lỗi nhanh chóng và chính xác hơn. Đây là hướng đi có cơ sở rõ ràng về mặt dữ liệu và phù hợp với xu thế nâng cao chất lượng sản phẩm trong ngành dệt may.

CHƯƠNG 4: PHÂN TÍCH VÀ THIẾT KẾ MÔ HÌNH

4.1 Cơ sở định hướng thiết kế mô hình

4.1.1 Sơ đồ quy trình phát triển mô hình học máy



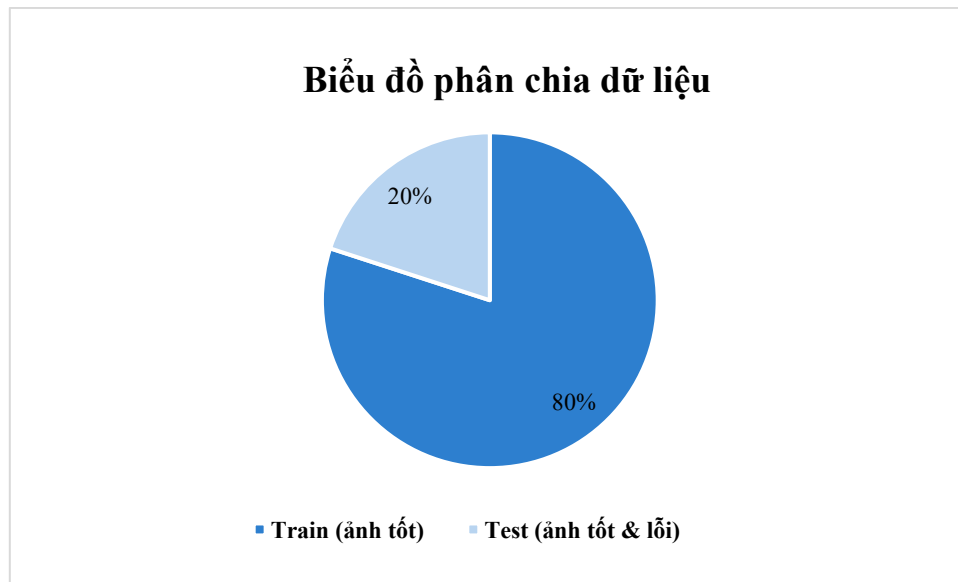
Hình 4.1: Sơ đồ quy trình phát triển mô hình học máy

4.1.2 Cơ sở định hướng thiết kế mô hình

Trong các bài toán tái tạo ảnh hoặc phát hiện bất thường không có nhãn, mô hình Autoencoder là một lựa chọn phổ biến nhờ khả năng học biểu diễn nén trong môi trường không giám sát. Trong bối cảnh phát hiện bất thường, mô hình được huấn luyện hoàn toàn trên tập ảnh không chứa lỗi, từ đó học được các đặc trưng phổ biến của dữ liệu bình thường. Khi xử lý một ảnh mới trong giai đoạn kiểm tra, nếu ảnh đó có chứa lỗi, mô hình sẽ không thể tái tạo chính xác các vùng bất thường do chưa từng được học qua các đặc trưng đó.

Sự khác biệt giữa ảnh gốc và ảnh tái tạo sẽ được biểu diễn dưới dạng sai số tái tạo (reconstruction error). Các vùng có sai số cao thường tương ứng với khu vực chứa dị thường trên ảnh đầu vào. Dựa vào đó, có thể xác định và khoanh vùng các khu vực bất thường một cách hiệu quả, ngay cả khi không có sẵn nhãn lỗi trong quá trình huấn luyện.

4.2 Thu thập dữ liệu



Hình 4.2: Biểu đồ phân chia dữ liệu

Dữ liệu sử dụng trong đề án được thu thập trực tiếp tại xưởng thành phẩm của nhà máy, nơi diễn ra các công đoạn kiểm tra và đóng gói sản phẩm sau quá trình sản xuất. Việc thu thập được tiến hành trong suốt thời gian công nhân làm việc, nhằm đảm bảo ghi nhận đầy đủ và chính xác các lỗi xuất hiện trong thực tế.

Số lượng dữ liệu trước khi xử lý: 5000 tấm ảnh, gồm ảnh có lỗi và ảnh tốt

Số lượng dữ liệu sau khi xử lý: 1250 tấm ảnh, gồm ảnh có lỗi và ảnh tốt

Các lỗi được ghi nhận thông qua phương pháp chụp ảnh trực tiếp khi phát hiện sản phẩm có dấu hiệu bất thường. Quá trình chụp được thực hiện tại hiện trường sản xuất, với mục tiêu thu thập hình ảnh ở nhiều góc độ khác nhau để tăng tính đa dạng và toàn diện cho bộ dữ liệu. Các loại lỗi phổ biến: nhảy sợi, kết đầu, sợi dọc thô, tạp chất, sợi ngang co rút,...

Trong bài báo của Gholamy et al. (2018) đã chỉ ra rằng tỉ lệ chia dữ liệu 80% cho huấn luyện và 20% cho kiểm tra là lựa chọn tối ưu, vì giúp giảm sai số ước lượng mô hình và đã được chứng minh là hiệu quả nhất cả về mặt thực nghiệm lẫn phân tích thống kê [17]. Do đó, sau khi thu thập, dữ liệu được tiền xử lý sau đó chia thành hai tập với tỷ lệ 80% cho huấn luyện (train) và 20% cho kiểm tra (test).

Bảng 4.1: Thông tin dữ liệu

Dữ liệu	Tên phân loại	Số lượng	Tỉ lệ (%)
Train	Ảnh tốt	1000	80
Test	Ảnh tốt + Ảnh lỗi	250	20

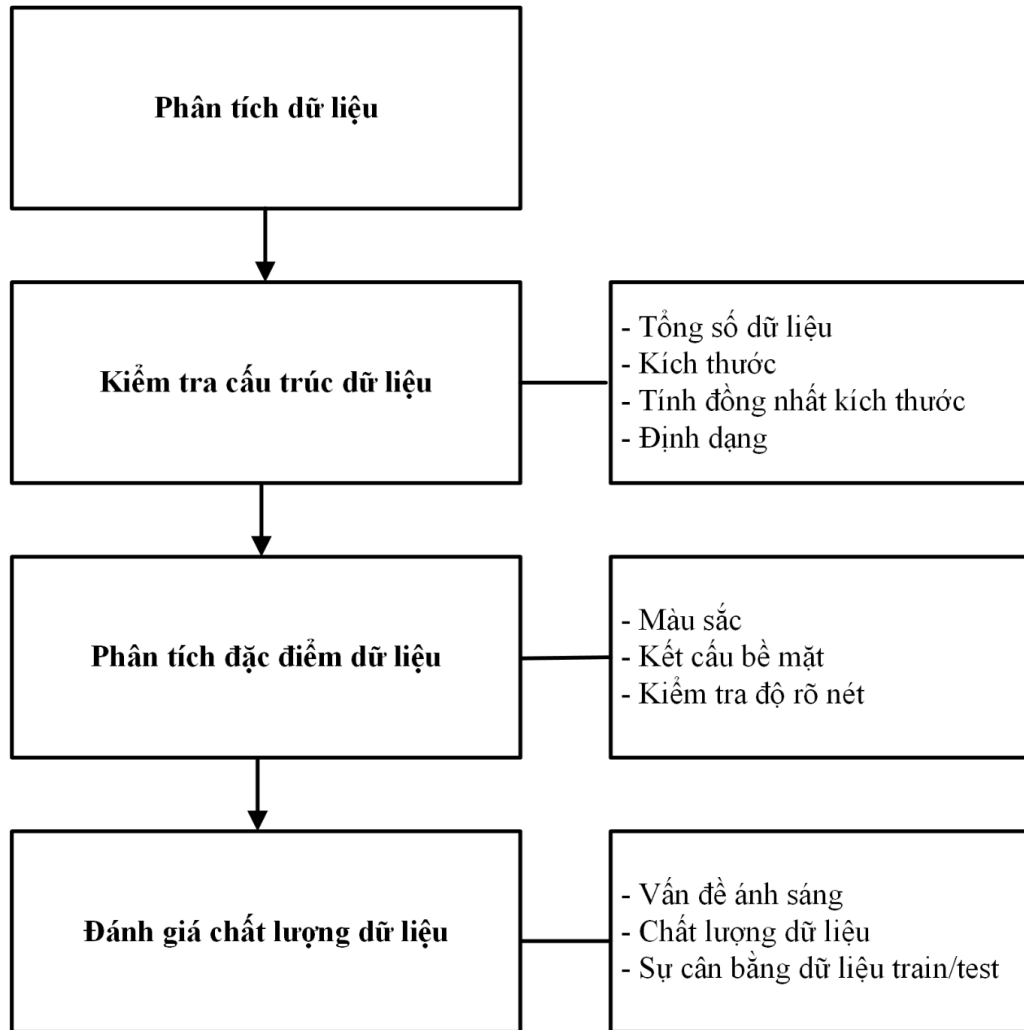
Bảng 4.2: Đánh giá chất lượng ảnh

Ảnh tốt	Ảnh lỗi
	

```
1 NUM_WORKERS = os.cpu_count()  
2 BATCH_SIZE = 32  
3 IMG_RESIZE = 256  
4 INPUT_CROP = 256  
5 TRAINING_SIZE = 1000  
6 TEST_SIZE = 250
```

Hình 4.3: Cấu hình thông số đầu vào cho quá trình huấn luyện

4.3 Phân tích dữ liệu đầu vào



Hình 4.4: Sơ đồ phân tích dữ liệu đầu vào

Bước phân tích dữ liệu được thực hiện nhằm đánh giá toàn diện đặc điểm, chất lượng và cấu trúc của tập dữ liệu hình ảnh vải denim trước khi huấn luyện mô hình Autoencoder. Qua đó đảm bảo dữ liệu được hiểu rõ để lựa chọn các phương pháp tiền xử lý phù hợp như chuẩn hóa, tăng cường dữ liệu, đồng thời phát hiện và khắc phục các vấn đề như nhiễu hoặc mất cân bằng dữ liệu nhằm tối ưu hóa quá trình huấn luyện, nâng cao khả năng tái tạo hình ảnh hoặc phát hiện khuyết tật trên vải denim một cách chính xác và hiệu quả. Quy trình này bao gồm các hoạt động chính như sau:

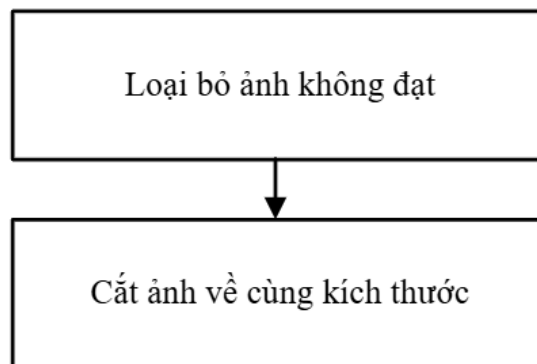
Kiểm tra cấu trúc dữ liệu: Xác định tổng số hình ảnh, kích thước (256x256 pixel) và định dạng (JPG). Đồng thời, kiểm tra tính đồng nhất về kích thước và định dạng để chuẩn bị cho các bước tiền xử lý sau này.

Phân tích đặc điểm dữ liệu: Đánh giá các đặc trưng thị giác như màu sắc chủ đạo (xanh đậm hoặc xanh nhạt đặc trưng của denim), kết cấu bề mặt và sự hiện diện của khuyết tật (lấn tạp chất, kết đầu, lỗ thủng, ...).

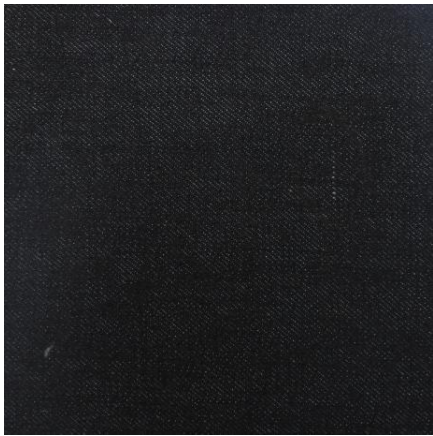
Đánh giá chất lượng dữ liệu: Phát hiện các vấn đề tiềm ẩn như nhiễu do ánh sáng không đồng đều, hình ảnh mờ hoặc các tệp bị hỏng. Ngoài ra, phân tích sự cân bằng giữa hai loại dữ liệu: hình ảnh vải bình thường và vải có khuyết tật, nhằm tránh hiện tượng mất cân bằng dữ liệu có thể gây ảnh hưởng đến hiệu suất mô hình.

4.4 Tiền xử lý dữ liệu

4.4.1 Tiền xử lý dữ liệu thô



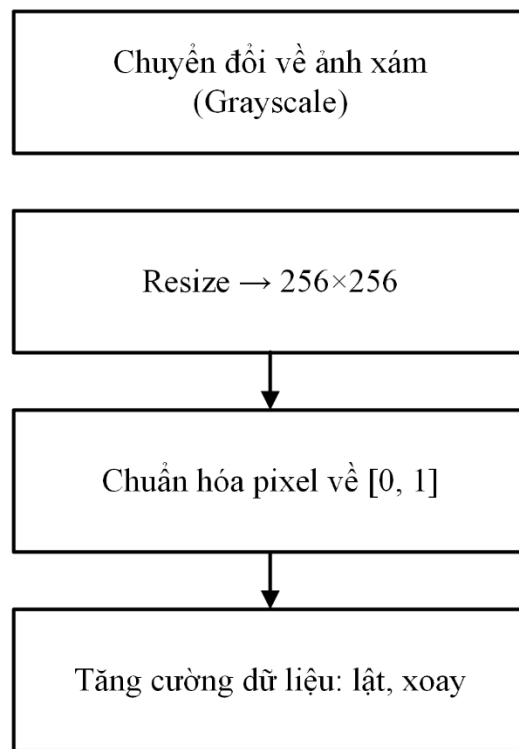
Hình 4.5: Sơ đồ tiền xử lý dữ liệu thô

Loại bỏ	Đạt chuẩn
	

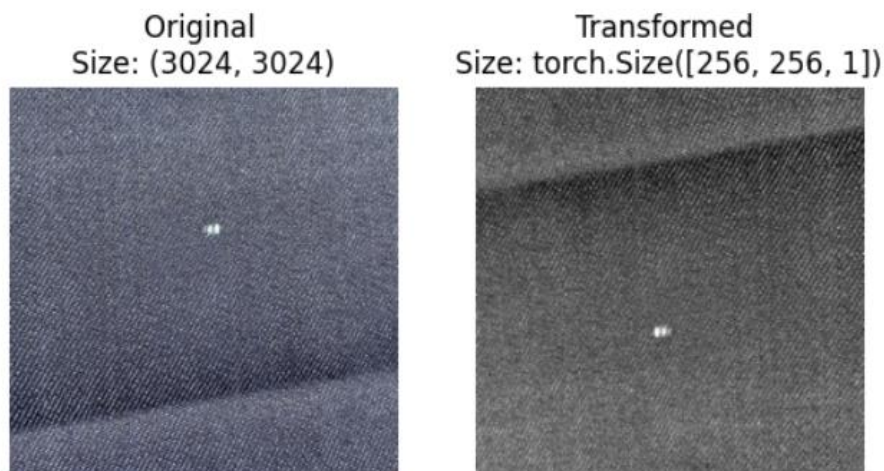
Hình 4.6: Nhận biết dữ liệu loại bỏ và dữ liệu đạt chuẩn

Đây là bước quan trọng trước khi tiến hành thiết kế mô hình. Trong giai đoạn này, các ảnh mờ, ảnh không rõ lỗi hoặc không hợp lệ sẽ được loại bỏ để đảm bảo chất lượng dữ liệu đầu vào. Sau đó, ảnh được cắt và điều chỉnh về cùng kích thước nhằm tạo sự nhất quán, thuận tiện cho quá trình huấn luyện và đánh giá mô hình.

4.4.2 Tiền xử lý dữ liệu trong mô hình



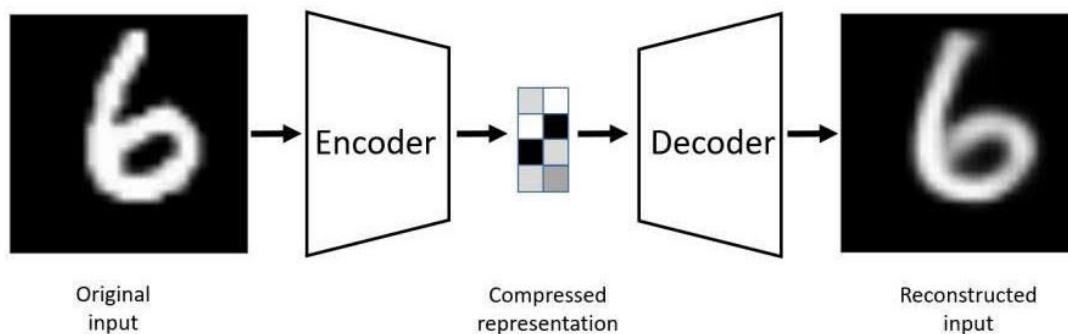
Hình 4.7: Sơ đồ tiền xử lý dữ liệu trong mô hình



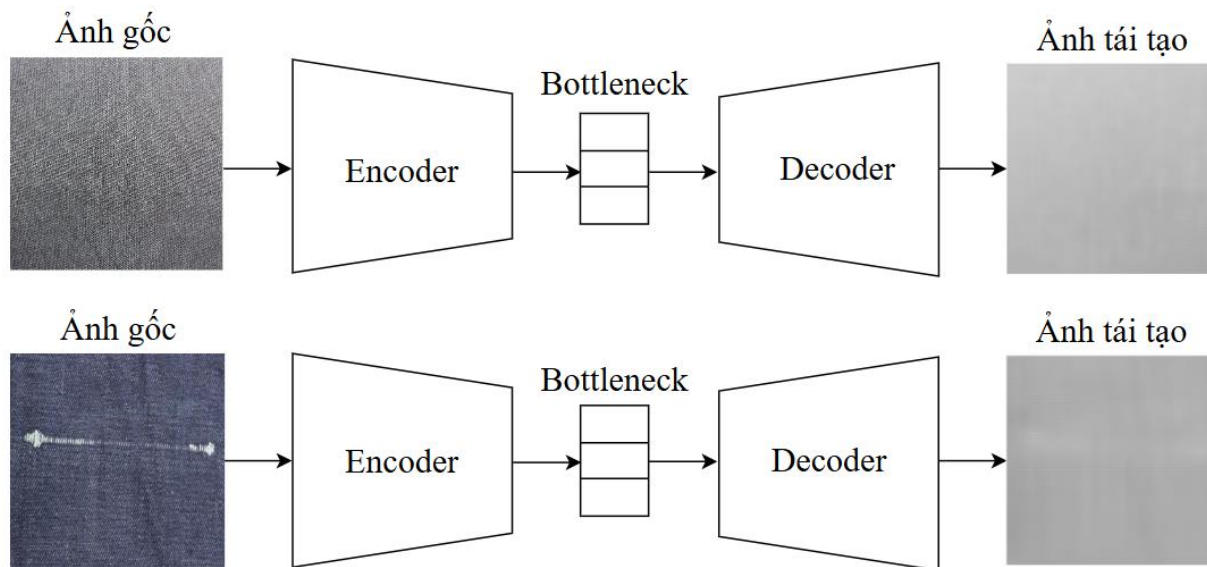
Hình 4.8: Ảnh gốc và ảnh đã chuyển đổi về dạng xám (Grayscale)

Trước khi đưa vào mô hình, toàn bộ dữ liệu ảnh được chuyển đổi về dạng ảnh xám (grayscale) nhằm giảm chiều dữ liệu và tập trung vào đặc trưng kết cấu thay vì màu sắc. Tiếp theo, ảnh được chuẩn hóa về dải giá trị $[0, 1]$ và resize về kích thước chuẩn 256×256 nếu chưa đạt yêu cầu. Đồng thời, các kỹ thuật tăng cường dữ liệu (augmentation) như xoay ảnh, lật ảnh theo chiều ngang hoặc dọc cũng được áp dụng nhằm giúp mô hình học tốt hơn và tăng khả năng tổng quát hóa. Điều này đặc biệt quan trọng trong bối cảnh học không giám sát, khi mô hình phải tự rút trích đặc trưng từ dữ liệu mà không có nhãn hướng dẫn.

4.5 Sơ đồ hệ thống xử lý dữ liệu



Hình 4.9: Sơ đồ hệ thống xử lý dữ liệu Autoencoder



Hình 4.10: Sơ đồ hệ thống xử lý dữ liệu Autoencoder thực

Quy trình xử lý ảnh vải denim được mô hình hóa thông qua kiến trúc Autoencoder, thể hiện dưới dạng một sơ đồ khối gồm ba giai đoạn chính: nén đặc trưng, tái tạo ảnh và đánh giá

sai số. Hệ thống này cho phép mô hình học cách biểu diễn và tái tạo lại dữ liệu hình ảnh một cách hiệu quả trong môi trường không giám sát.

Đầu vào (Input): bao gồm các ảnh vải denim thu nhận từ hệ thống camera công nghiệp hoặc thiết bị ghi hình. Ảnh đầu vào đại diện cho thông tin trực quan về bề mặt vải, được sử dụng làm dữ liệu gốc cho quá trình học.

Bộ mã hóa (encoder): thực hiện trích xuất và nén các đặc trưng chính của ảnh thành một biểu diễn trung gian có chiều thấp (latent vector). Quá trình này loại bỏ các thông tin dư thừa, chỉ giữ lại những yếu tố đặc trưng mang tính đại diện cho cấu trúc và kết cấu của bề mặt vải.

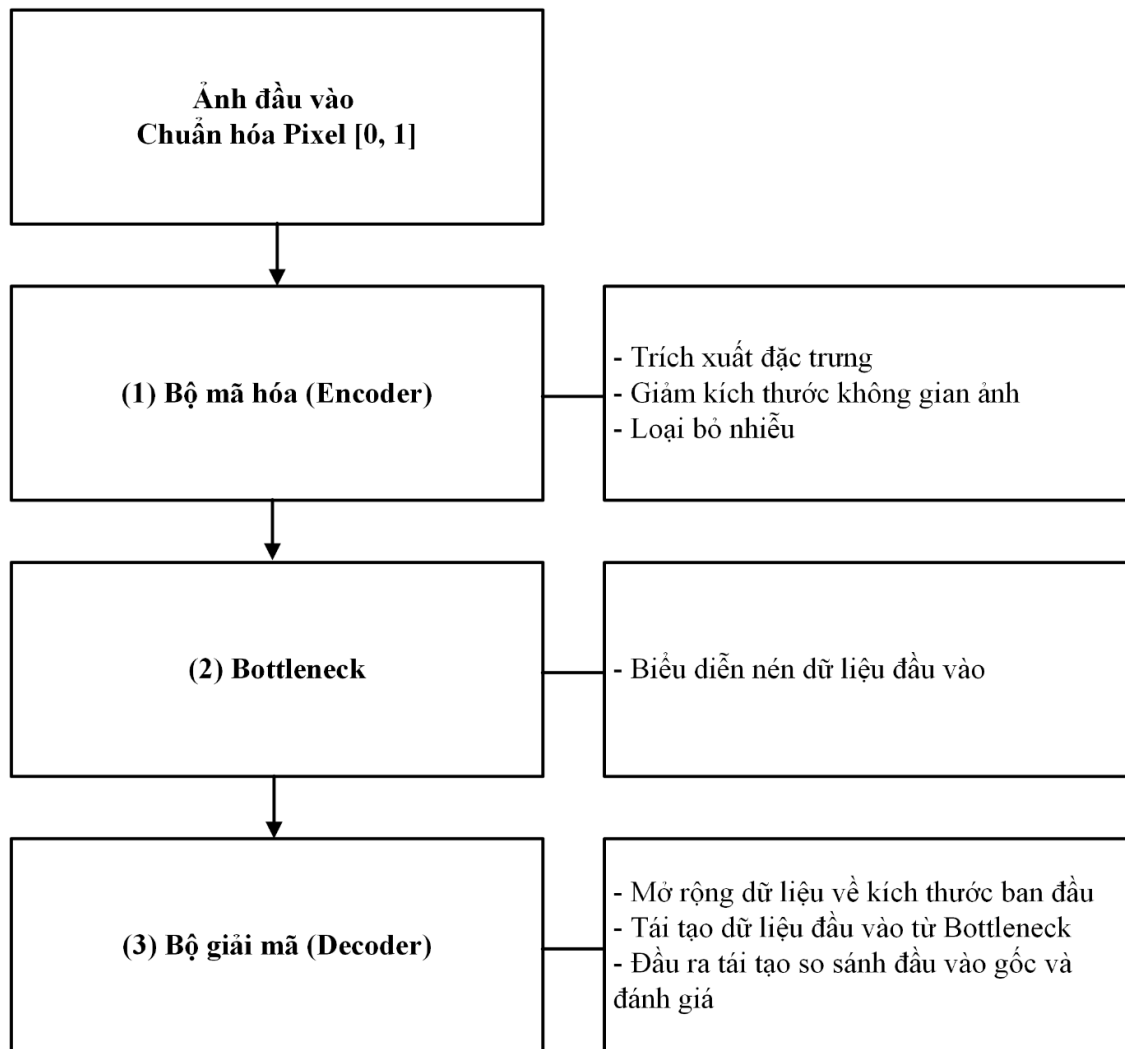
Bộ giải mã (Decoder): sử dụng biểu diễn trung gian này để tái tạo lại ảnh gốc. Mục tiêu của giai đoạn này là khôi phục hình ảnh đầu vào một cách tối ưu nhất có thể, dựa trên thông tin đã được mã hóa.

Cuối cùng, hệ thống so sánh ảnh tái tạo với ảnh gốc để tính toán sai số tái tạo. Chênh lệch giữa hai ảnh phản ánh mức độ mất mát thông tin trong quá trình mã hóa - giải mã. Dựa trên sai số này, mô hình cập nhật trọng số qua quá trình lan truyền ngược nhằm cải thiện hiệu suất tái tạo trong các vòng huấn luyện tiếp theo.

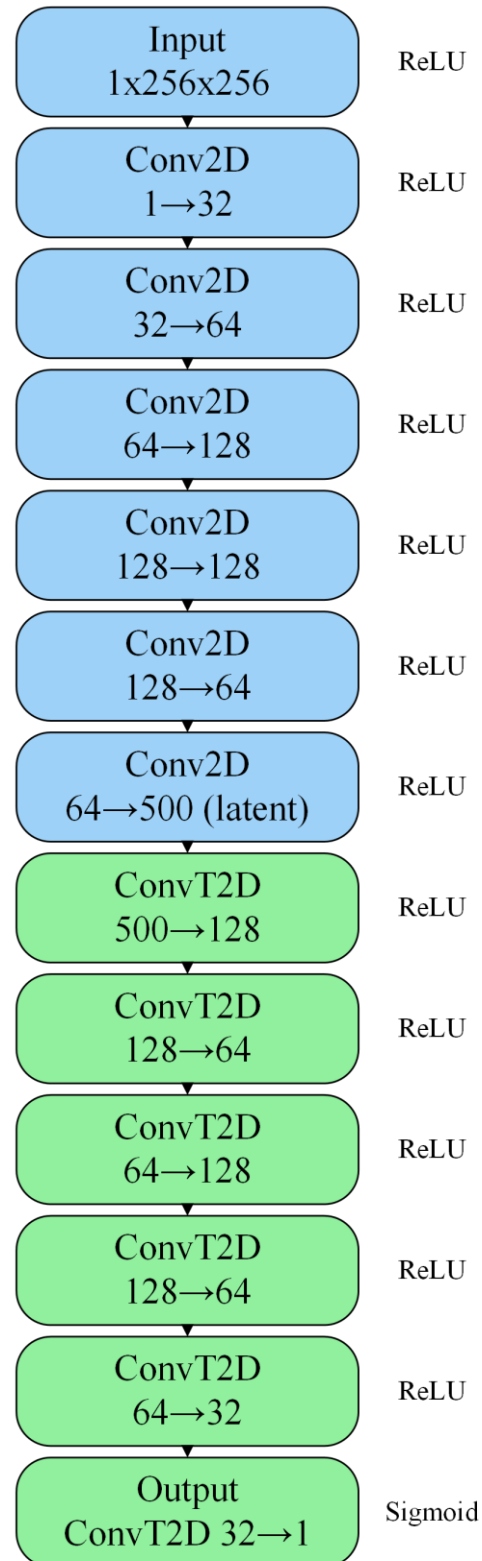
Kiến trúc này cho phép mô hình học được các đặc trưng quan trọng của ảnh vải mà không cần nhãn, đồng thời hỗ trợ hiệu quả cho các ứng dụng như phát hiện bất thường trên bề mặt vải denim.

4.6 Thiết kế kiến trúc mô hình

4.6.1 Kiến trúc tổng thể của mô hình



Hình 4.11: Sơ đồ kiến trúc tổng thể của mô hình



Hình 4.12: Cấu trúc Autoencoder với các lớp Conv2D và ConvTranspose2D

(1) Mô hình được thiết kế theo kiến trúc Autoencoder gồm hai thành phần chính: bộ mã hóa (encoder) và bộ giải mã (decoder). Mục tiêu của mô hình là học một biểu diễn nén của ảnh đầu vào và tái tạo lại ảnh đó với độ sai lệch nhỏ nhất có thể. Ảnh đầu vào có kích thước $1 \times 256 \times 256$ (ảnh xám), được đưa qua chuỗi các lớp tích chập để trích xuất đặc trưng, sau đó được giải mã thông qua các lớp tích chập chuyển vị để tái tạo lại ảnh đầu ra.

(2) Bộ mã hóa (Encoder) bao gồm sáu lớp tích chập 2 chiều (Conv2D) với số lượng kênh tăng dần: từ $1 \rightarrow 32 \rightarrow 64 \rightarrow 128 \rightarrow 128 \rightarrow 64 \rightarrow 500$. Mỗi lớp được nối tiếp với hàm kích hoạt ReLU để tăng tính phi tuyến trong quá trình học. Quá trình này không chỉ giúp mô hình học được các đặc trưng cục bộ như cạnh, kết cấu mà còn giảm dần độ phân giải của ảnh, dẫn đến biểu diễn nén cuối cùng (latent vector) với 500 kênh. Đây là nơi lưu trữ thông tin cốt lõi của ảnh và sẽ được sử dụng làm đầu vào cho bộ giải mã.

(3) Bộ giải mã (Decoder) sử dụng các lớp tích chập chuyển vị 2 chiều (ConvT2D) để phục hồi ảnh từ biểu diễn nén. Các lớp này thực hiện đồng thời hai nhiệm vụ: tăng kích thước không gian và điều chỉnh số lượng kênh đặc trưng. Kiến trúc cụ thể bao gồm chuỗi chuyển đổi: $500 \rightarrow 128 \rightarrow 64 \rightarrow 128 \rightarrow 64 \rightarrow 32 \rightarrow 1$. Tất cả các lớp đều sử dụng hàm kích hoạt ReLU, ngoại trừ lớp cuối cùng dùng hàm Sigmoid để chuẩn hóa đầu ra về khoảng $[0, 1]$, phù hợp với ảnh xám đã được chuẩn hóa.

Việc tăng số kênh tạm thời từ 64 lên 128 rồi quay lại 64 ($64 \rightarrow 128 \rightarrow 64$) không phải là dư thừa mà mang tính chiến lược, nhằm mở rộng không gian biểu diễn tạm thời, cho phép mô hình học thêm các đặc trưng phi tuyến phức tạp hơn. Trong học sâu, các lớp trung gian với số kênh lớn hơn thường đóng vai trò như các tầng xử lý đặc trưng ngữ cảnh, giúp tái cấu trúc lại các chi tiết bị mất trong quá trình nén. Sau đó, số kênh được giảm trở lại để phù hợp với kiến trúc phân tầng, hướng đến cấu trúc đầu ra cuối cùng.

4.6.2 Tham số mô hình

```
-----  
Total params: 2,006,325  
Trainable params: 2,006,325  
Non-trainable params: 0  
Total mult-adds (Units.GIGABYTES): 52.91  
=====
```

Input size (MB):	8.39
Forward/backward pass size (MB):	503.73
Params size (MB):	8.03
Estimated Total Size (MB):	520.14

```
=====
```

Hình 4.13: Tham số mô hình

Với tổng cộng chỉ khoảng 2 triệu tham số, mô hình có quy mô nhỏ hơn đáng kể so với các mạng học sâu hiện đại như ResNet-50 (~25 triệu tham số) hay EfficientNet-B0 (~5 triệu tham số). Theo nghiên cứu của Tan & Le (2019), việc tối ưu hóa đồng thời chiều sâu, chiều rộng và độ phân giải đầu vào của mô hình có thể giúp các mạng nhỏ hơn đạt được hiệu suất cao với chi phí tính toán thấp hơn — một yếu tố then chốt để triển khai trên thiết bị hạn chế tài nguyên [18].

Thêm vào đó, mô hình chỉ sử dụng khoảng 520 MB bộ nhớ GPU cho một lượt huấn luyện hoàn chỉnh, bao gồm cả quá trình lan truyền tiến và lan truyền ngược. Đây là mức tiêu thụ thấp hơn nhiều so với các mô hình thị giác máy tính thông dụng hiện nay, cho phép triển khai hiệu quả trên GPU tầm trung.

Đáng chú ý, trong các bài toán như phát hiện lỗi bề mặt vải, bảng mạch hoặc sản phẩm công nghiệp, mô hình lý tưởng thường cần đảm bảo: (1) nhẹ, để chạy thời gian thực; (2) chính xác cao, để nhận ra sai lệch nhỏ và (3) tiêu thụ tài nguyên thấp, để phù hợp với dây chuyền sản xuất. Theo Bergmann et al. (2018), các mô hình autoencoder đơn giản nhưng được huấn luyện với hàm mất mát phù hợp như SSIM vẫn có thể đạt hiệu suất AUC lên đến 0.966 trong nhiệm vụ phát hiện lỗi mà không cần sử dụng mạng sâu hoặc kiến trúc phức tạp [15].

4.7 Các công thức toán học liên quan

4.7.1 Tính kích thước đầu ra sau Conv/Deconv

Khi xây dựng mô hình tích chập (CNN), việc đảm bảo tính toán đúng kích thước đầu ra là điều kiện tiên quyết để khớp kiến trúc. Kích thước đầu ra của mỗi lớp tích chập hoặc lớp chuyển vị tích chập được tính bằng công thức:

$$H_{out} = \left\lfloor \frac{H_{in} + 2p - k}{s} \right\rfloor + 1 \quad (4.1)$$

$$W_{out} = \left\lfloor \frac{W_{in} + 2p - k}{s} \right\rfloor + 1 \quad (4.2)$$

Trong đó:

H_{in}, W_{in} : chiều cao và chiều rộng đầu vào

k: kích thước kernel

s: bước nhảy (stride)

p: padding

H_{out}, W_{out} : chiều cao và chiều rộng đầu ra

4.7.2 Tính số lượng tham số từng lớp

Số lượng tham số huấn luyện của một lớp Conv2D hoặc ConvTranspose2D được tính theo công thức:

$$Params = (C_{in} \cdot k \cdot k + b) \cdot C_{out} \quad (4.3)$$

Trong đó:

C_{in} : số kênh đầu vào

C_{out} : số kênh đầu ra

k: kích thước kernel

b: số bias (thường là 1 nếu có bias, 0 nếu không)

4.7.3 Tính tổng số tham số huấn luyện toàn mô hình

Tổng số tham số huấn luyện của mô hình được tính bằng cách cộng toàn bộ các tham số từ các lớp có huấn luyện (learnable layers):

$$\text{Tổng số tham số} = \sum_{l=1}^L (C_{in}^{(l)} \cdot k_l \cdot k_l + b_l) \cdot C_{out}^{(l)} \quad (4.4)$$

Trong đó:

L: tổng số lớp có huấn luyện trong mô hình

$C_{in}^{(l)}, C_{out}^{(l)}$: số kênh vào và ra của lớp thứ l

k_l : kích thước kernel lớp thứ l

b_l : số bias của lớp thứ l

4.8 Huấn luyện mô hình

4.8.1 Hàm mất mát sử dụng trong mô hình

4.8.1.1 MS-SSIM loss

Được sử dụng để đo lường sự sai khác về mặt cấu trúc giữa ảnh gốc và ảnh tái tạo. Giá trị MS-SSIM vốn dĩ là một độ đo tương đồng (similarity), do đó để sử dụng làm hàm mất mát, ta tính theo công thức 1 - MS_SSIM. Cách làm này giúp chuyển đổi từ độ đo tương đồng sang độ đo sai khác, từ đó mô hình có thể tối ưu theo hướng giảm thiểu sự sai khác cấu trúc giữa hai ảnh. Nhờ đó, ảnh tái tạo giữ được các đặc trưng hình học và bố cục tổng thể giống với ảnh ban đầu [19].

$$\text{MS_SSIM}(x, \hat{x}) = l_M(x, \hat{x}) \prod_{j=1}^M cs_j(x, \hat{x}) \quad (4.5)$$

$$L_{\text{MS_SSIM}} = 1 - \text{MS_SSIM}(x, \hat{x}) \quad (4.6)$$

$l_M(x, \hat{x})$: Thành phần độ sáng (luminance) ở tầng (scale) cuối cùng. Đo sự tương đồng về độ sáng tổng thể giữa hai ảnh ở mức thô nhất (coarsest scale).

$cs_j(x, \hat{x})$: Thành phần độ tương phản và cấu trúc (contrast-structure) tại tầng thứ j

$\prod_{j=1}^M cs_j(x, \hat{x})$: Tích của tất cả các thành phần tương phản - cấu trúc ở các tầng từ 1 đến M. Tầng tính tổng quát hóa kết cấu và mức độ chi tiết bằng cách tính từ nhiều độ phân giải ảnh.

4.8.1.2 Edge Loss

Để bảo toàn các cạnh (edges) trong ảnh, mô hình sử dụng toán tử Sobel để trích xuất biên của cả ảnh gốc và ảnh tái tạo. Kết quả thu được là hai ảnh thể hiện vị trí và cường độ cạnh. Sau đó, ta tính sai số L1 (Mean Absolute Error) giữa hai ảnh này để đo mức độ lệch biên. Cơ chế này giúp mô hình tái tạo chính xác hơn các đường viền và chi tiết ranh giới - những vùng thường mang thông tin quan trọng trong việc phát hiện lỗi sản phẩm [20].

$$L_{\text{edge}} = \frac{1}{N} \sum_{i=1}^N |S(x_i) - S(\hat{x}_i)| \quad (4.7)$$

$S(x_i)$: cường độ biên tại pixel i của ảnh gốc

$S(\hat{x}_i)$: cường độ biên tại pixel i của ảnh tái tạo

N : tổng số điểm ảnh

4.8.1.3 Texture loss

Nhằm duy trì sự nhất quán về kết cấu giữa ảnh gốc và ảnh tái tạo. Đầu tiên, ảnh được làm mờ bằng bộ lọc Gaussian để loại bỏ các chi tiết nhỏ và cạnh sắc, chỉ giữ lại cấu trúc kết cấu tổng thể. Sau đó, trung bình không gian của ảnh được trừ ra nhằm loại bỏ độ chênh lệch về độ sáng, giúp tập trung vào phần biến thiên kết cấu. Cuối cùng, sai số bình phương trung bình (MSE) giữa hai ảnh làm mờ được sử dụng để tính toán mức độ khác biệt về kết cấu. Điều này đặc biệt quan trọng với các hình ảnh có họa tiết dạng lặp như vải hoặc bề mặt vật liệu [21].

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G_{ij}^l - \hat{G}_{ij}^l)^2 \quad (4.8)$$

E_l : texture loss tại tầng l

N_l^2 : số kênh đặc trưng (feature maps) tại tầng l

M_l^2 : số pixel trên mỗi feature map (tức width x height)

G_{ij}^l : phần tử i,j , j,i của Gram matrix từ ảnh gốc

$\hat{G}_{ij}^l$: phần tử tương ứng từ ảnh tái tạo

4.8.1.4 Pixel Loss

Được tính đơn giản bằng sai số L1 giữa từng cặp điểm ảnh tương ứng trong ảnh gốc và ảnh tái tạo. Mặc dù là một hàm mất mát cơ bản, nó vẫn đóng vai trò thiết yếu trong việc giữ cho

ảnh tái tạo không bị lệch quá xa về mặt giá trị pixel. Nó cũng hỗ trợ tạo sự ổn định trong quá trình học, đồng thời hỗ trợ cho các thành phần mất mát phức tạp hơn như MS-SSIM, biên và kết cấu [19], [22].

$$L_{l1} = \frac{1}{N} \sum_{i \in 1}^N |x(i) - \hat{x}(i)| \quad (4.9)$$

$L_{l1}(P)$: giá trị sai số trung bình giữa ảnh tái tạo và ảnh gốc

N : Là tổng số điểm ảnh (pixel) trong ảnh hoặc patch đó

$x(i)$: Giá trị pixel tại vị trí i trong ảnh tái tạo

$\hat{x}(i)$: Giá trị pixel tại vị trí I trong ảnh gốc

4.8.1.5 Loss kết hợp và công thức tổng quát

Tổng loss được xây dựng từ sự kết hợp có trọng số giữa L1 Loss, MS-SSIM, Edge Loss và Texture Loss, nhằm cân bằng giữa độ chính xác số học và cảm nhận thị giác

$$L_{\text{total}} = \lambda_1 \cdot L_{\text{MS-SSIM}} + \lambda_2 \cdot L_{\text{edge}} + \lambda_3 \cdot L_{\text{texture}} + \lambda_4 \cdot L_{\text{pixel}} \quad (4.10)$$

λ_i : hệ số điều chỉnh cho từng hàm mất mát

4.8.2 Nguyên lý hoạt động của mô hình

Mô hình được huấn luyện theo hướng không giám sát, chủ yếu dựa trên tập dữ liệu gồm các ảnh "tốt" không chứa lỗi. Trong quá trình huấn luyện, mô hình học cách mã hóa và tái tạo lại các ảnh này sao cho hình ảnh đầu ra càng giống đầu vào càng tốt. Do chưa từng tiếp xúc với ảnh lỗi, mô hình có xu hướng tái tạo kém các đặc trưng bất thường (như lỗ thủng, đứt sợi, tạp chất...), dẫn đến sai số tái tạo cao tại các vị trí có lỗi.

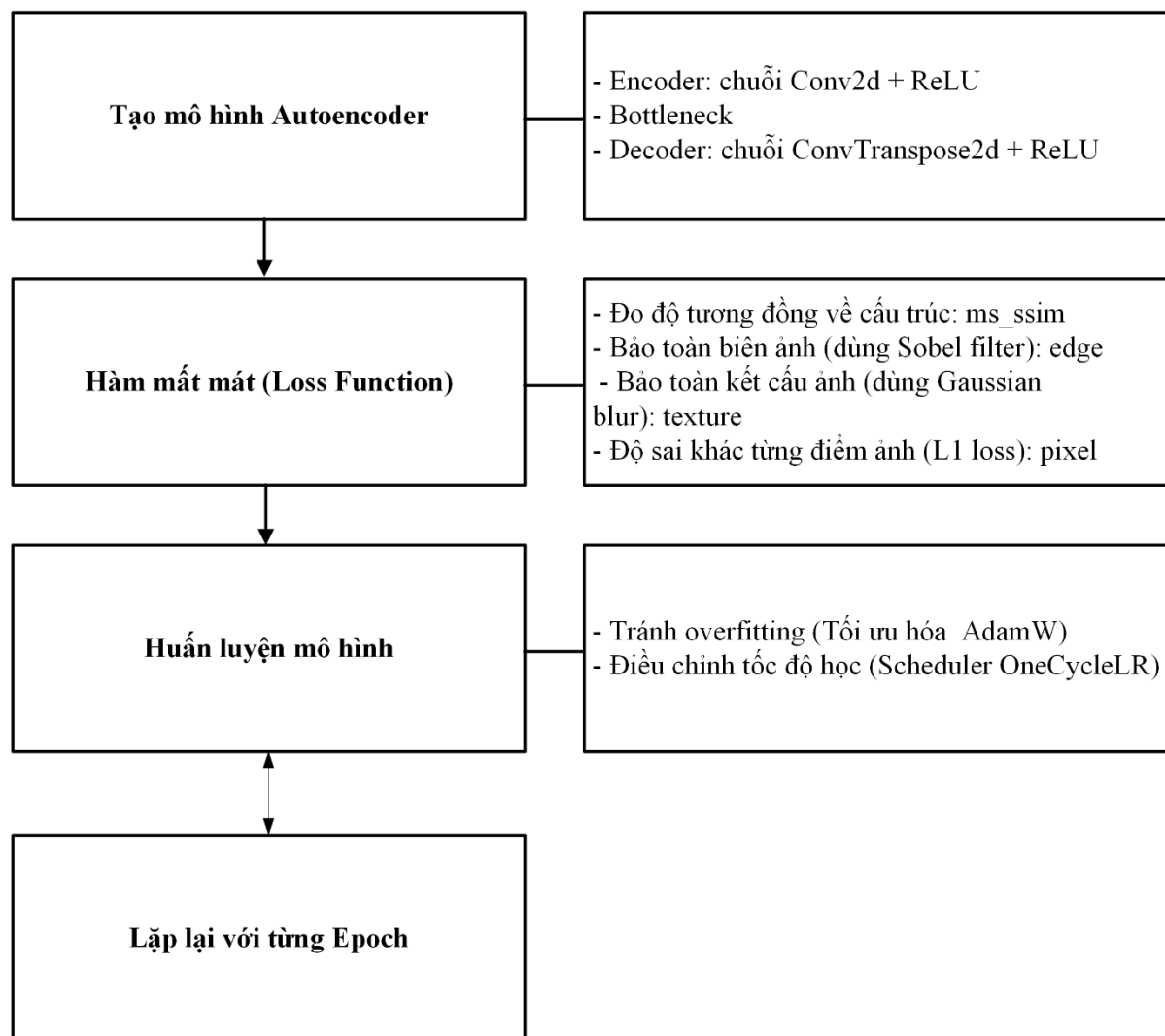
Từ nguyên lý hoạt động đó, cơ chế phát hiện lỗi được xây dựng dựa trên việc so sánh sự khác biệt giữa ảnh gốc và ảnh tái tạo: nếu sai số thấp, ảnh được xem là bình thường; nếu sai số cao, có khả năng là ảnh lỗi. Theo Torabi et al. (2023), độ sai khác này thường được đo bằng các chỉ số như L2 loss, SSIM (Structural Similarity Index) hoặc Max Difference. Tuy nhiên, để lựa chọn chỉ số đánh giá phù hợp, cần xét đến đặc điểm của môi trường ứng dụng thực tế [23].

Trong môi trường sản xuất thực tế, ảnh chụp bề mặt vải thường dễ bị nhiễu do ánh sáng không đồng đều, kết cấu bề mặt thay đổi và chuyển động nhẹ trong dây chuyền sản xuất. Những yếu tố này làm cho các chỉ số đánh giá dựa trên cấu trúc như SSIM trở nên nhạy

cảm với nhiễu nhỏ, dễ tạo ra sai số dù ảnh không chứa lỗi, từ đó dẫn đến việc đánh giá sai lệch vùng lỗi và làm giảm độ chính xác của hệ thống [23].

Chính vì vậy, việc sử dụng chỉ số “Max Difference” (sai khác cực đại) mang lại hiệu quả cao hơn trong ngữ cảnh này. Khác với SSIM vốn đánh giá toàn cục dựa trên tương quan cấu trúc và độ tương phản, Max Difference chỉ tập trung vào điểm lệch lớn nhất giữa ảnh gốc và ảnh tái tạo. Cách tiếp cận này giúp mô hình nhấn mạnh vào các bất thường rõ ràng mà mô hình không thể tái tạo - tức là lỗi thực sự, đồng thời bỏ qua các sai lệch nhỏ không mang tính chất khuyết tật, qua đó tăng độ chính xác trong phát hiện lỗi trên sản phẩm vải.

4.8.3 Cấu hình huấn luyện



Hình 4.14: Cấu hình huấn luyện

```
self.weights = {  
    'ms_ssim': 0.4,  
    'edge': 0.25,  
    'texture': 0.25,  
    'pixel': 0.1
```

Hình 4.15: Trọng số huấn luyện

Để nâng cao chất lượng tái tạo hình ảnh và hỗ trợ mô hình học được các đặc trưng quan trọng, một hàm mất mát tùy chỉnh đã được xây dựng. Đây là một hàm tổ hợp nhiều thành phần đánh giá chất lượng tái tạo ảnh không chỉ dựa trên sự tương đồng điểm ảnh mà còn bao gồm các yếu tố về cấu trúc, biên cạnh và kết cấu. Cụ thể, hàm mất mát bao gồm 4 thành phần chính, với trọng số được định nghĩa trước:

Bảng 4.3: Trọng số của từng thành phần

Thành phần	Trọng số	Mục đích
MS-SSIM Loss	0.4	Đo mức độ tương đồng cấu trúc hình ảnh ở nhiều tỉ lệ (multi-scale), phản ánh độ bảo toàn cấu trúc tổng thể.
Edge Loss	0.25	Đảm bảo các biên cạnh trong ảnh tái tạo không bị mất mát hoặc làm mờ, thông qua hàm lọc Sobel.
Texture Loss	0.25	Giữ nguyên tính đồng nhất kết cấu bằng cách so sánh ảnh đã làm mờ Gaussian giữa ảnh gốc và ảnh tái tạo.
Pixel Loss	0.1	So sánh trực tiếp điểm ảnh gốc và tái tạo (mặc dù có trọng số thấp để tránh chi phối).

MS-SSIM Loss: chỉ số đánh giá độ tương đồng hình ảnh dựa trên cấu trúc thị giác đa cấp độ. Trọng số (0.4) được gán cho MS-SSIM loss vì nó phản ánh tốt hơn cách con người cảm nhận sự khác biệt giữa hai ảnh, đặc biệt là trong các bài toán phát hiện lỗi, nơi những khác biệt nhỏ về cấu trúc thường là dấu hiệu của bất thường. Theo Wang et al. (2003), MS-SSIM được chứng minh là có tương quan cao với đánh giá chủ quan của con người về chất lượng ảnh, vượt trội so với các chỉ số như PSNR hay MSE. Không chỉ là một chỉ số đánh giá, MS-SSIM còn được sử dụng làm thành phần chính trong hàm mất mát để huấn luyện mô hình tái tạo ảnh [24]. Bergmann et al. (2018) chứng minh rằng việc dùng SSIM thay cho

MSE trong huấn luyện autoencoder giúp mô hình nhận diện lỗi tốt hơn đáng kể [15]. Bên cạnh đó, Zhao et al. (2018) trong bài nghiên cứu "Loss Functions for Image Restoration with Neural Networks" cũng đã chỉ ra rằng MS-SSIM không chỉ giúp mô hình học được chi tiết cấu trúc mà còn làm giảm hiện tượng mờ (blurring) so với MSE [19].

Với các dẫn chứng trên, có thể thấy MS-SSIM là một thành phần loss rất phù hợp để sử dụng trọng số lớn, giúp mô hình học hiệu quả hơn trong việc bảo toàn cấu trúc ảnh và phát hiện sai lệch bất thường.

Edge Loss: đóng vai trò quan trọng trong việc nhấn mạnh các khu vực có cạnh và ranh giới rõ nét, vốn là nơi thường xuất hiện các khuyết tật như vết nứt, vết cắt hoặc thay đổi cấu trúc bề mặt. Trong bài toán tái tạo ảnh để phát hiện lỗi, nếu mô hình không học tốt các biên, lỗi có thể bị làm mờ hoặc mất chi tiết trong quá trình tái tạo. Do đó, việc gán trọng số (0.25) cho Edge Loss là hợp lý để đảm bảo mô hình tái tạo chính xác các ranh giới, đồng thời phát hiện lỗi rõ ràng hơn. Liu et al. (2018), đưa thêm thành phần Edge-aware Loss để tăng khả năng phục hồi chi tiết viền hình ảnh, đạt kết quả vượt trội trong bài toán khôi phục ảnh bị che lấp [25].

Texture Loss: đóng vai trò giúp mô hình học và tái tạo lại chi tiết kết cấu bề mặt - bao gồm các vân nhỏ, họa tiết lặp, hoặc bất thường cục bộ. Trong các bài toán như phát hiện lỗi vải, vật liệu in hay kiểm tra chất lượng sản phẩm công nghiệp, lỗi thường làm gián đoạn tính đồng nhất của kết cấu, trong khi giá trị pixel tổng thể có thể không thay đổi nhiều. Do đó, Texture Loss trở thành một thành phần thiết yếu để mô hình phát hiện được các sai lệch tinh vi trong ảnh. Việc đặt trọng số 0.25 cho Texture Loss là một lựa chọn cân bằng hợp lý: không quá cao để tránh việc mô hình trở nên quá nhạy với các nhiễu nhỏ hoặc sai lệch không quan trọng, nhưng cũng không quá thấp để đảm bảo rằng mô hình vẫn học được các khác biệt kết cấu tinh vi - những điểm mà các hàm mất mát dựa trên pixel hoặc SSIM có thể bỏ sót.

Pixel Loss: Giúp duy trì độ chính xác cơ bản giữa ảnh gốc và ảnh tái tạo, nhưng nếu được gán trọng số quá cao, mô hình dễ tạo ra ảnh bị mờ và thiếu chi tiết. Dosovitskiy và Brox (2016) cảnh báo rằng “việc chỉ dùng pixel-wise loss như L2 thường tạo ra ảnh tái dựng bị mờ và thiếu chi tiết tần số cao” [26]. Zhao et al. (2018) cũng chỉ ra rằng pixel loss “không tương quan tốt với nhận thức thị giác và dễ gây hiện tượng mượt hóa quá mức” [19].

Vì vậy, trọng số 0.1 được lựa chọn để giữ vai trò hỗ trợ - đảm bảo độ chính xác tổng thể mà không làm lu mờ các thành phần quan trọng hơn như SSIM hay texture loss.

```
1 optimizer = torch.optim.AdamW(  
2     params=model.parameters(),  
3     lr=1e-3,  
4     betas=(0.9, 0.999),  
5     weight_decay=1e-5  
6 )  
7  
8 # Learning Rate Scheduler  
9 scheduler = torch.optim.lr_scheduler.OneCycleLR(  
10     optimizer,  
11     max_lr=3e-3,  
12     steps_per_epoch=len(train_dataloader),  
13     epochs=NUM_EPOCHS,  
14     pct_start=0.3,  
15     anneal_strategy='cos'  
16 )
```

Hình 4.16: Tối ưu hóa AdamW và điều chỉnh tốc độ học

Trong quá trình huấn luyện, mô hình sử dụng thuật toán tối ưu AdamW - một biến thể của Adam được thiết kế để cải thiện hiệu quả regularization thông qua việc tách biệt thuật toán cập nhật tham số và điều chỉnh trọng số phạt (weight decay). Cấu hình cụ thể gồm tốc độ học ban đầu $lr = 1e-3$, các hệ số động lượng $betas = (0.9, 0.999)$ và hệ số phạt $weight_decay = 1e-5$. Sự kết hợp này nhằm đảm bảo quá trình tối ưu diễn ra ổn định và hạn chế overfitting.

Bên cạnh đó, mô hình áp dụng bộ điều chỉnh tốc độ học OneCycleLR nhằm kiểm soát tốc độ học theo chu kỳ. Cụ thể, tốc độ học được tăng dần đến giá trị cực đại $max_lr = 3e-3$ trong giai đoạn đầu ($pct_start = 0.3$) và sau đó giảm dần theo chiến lược cosine annealing ($anneal_strategy = 'cos'$). Cách điều chỉnh này giúp mô hình tiếp cận không gian tham số hiệu quả hơn, đồng thời cải thiện khả năng hội tụ của mô hình.

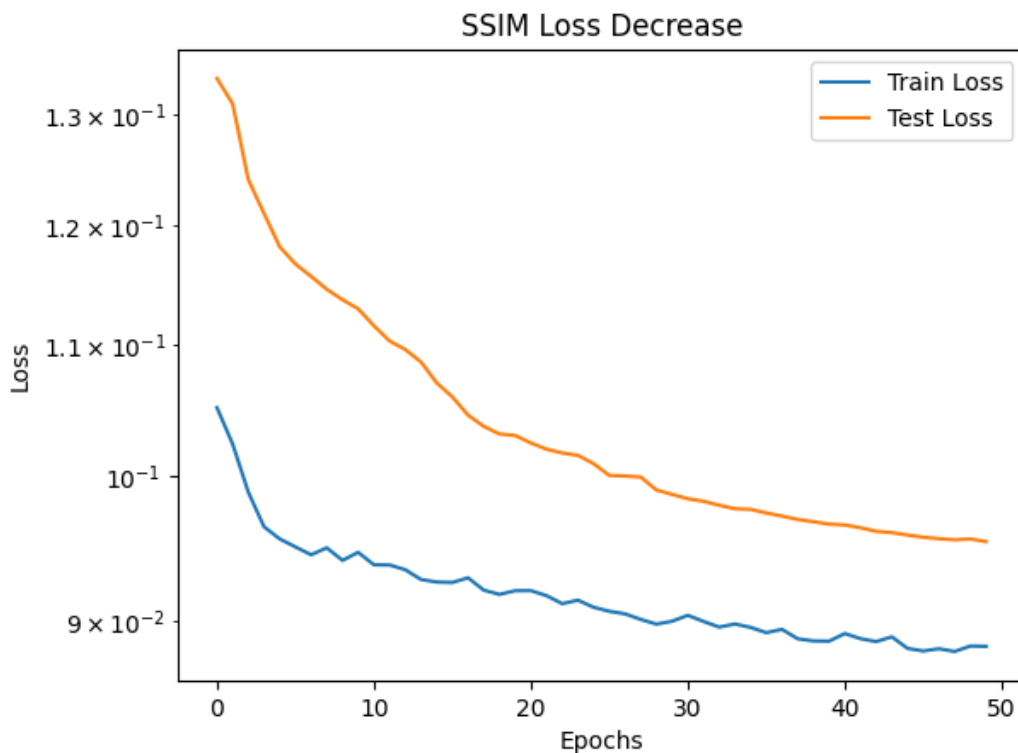
Việc sử dụng kết hợp AdamW và OneCycleLR trong cấu hình này được thiết lập với mục tiêu cân bằng giữa tốc độ hội tụ và khả năng tổng quát hóa của mô hình trong bài toán tái tạo ảnh và phát hiện bất thường.

```
10 train(  
11     model=model,  
12     train_dataloader=train_dataloader,  
13     val_dataloader=test_dataloader,  
14     optimizer=optimizer,  
15     loss_fn=criterion,  
16     device=device,  
17     num_epochs=50,  
18     save_path="best_model.pth"  
19 )
```

Hình 4.17: Huấn luyện mô hình

Quá trình huấn luyện mô hình với các thành phần chính bao gồm: kiến trúc mô hình (truyền vào qua tham số model), bộ dữ liệu huấn luyện (train_dataloader), bộ dữ liệu kiểm tra (test_dataloader), trình tối ưu hóa (optimizer), hàm mất mát (loss_fn) và số lượt huấn luyện (NUM_EPOCHS). Quá trình huấn luyện được thực hiện thông qua hàm train(), giúp mô hình học từ dữ liệu và đánh giá hiệu suất qua từng epoch.

4.8.4 Hiệu suất huấn luyện



Hình 4.18: Hiệu suất huấn luyện

Training: 100%

50/50 [03:58<00:00, 4.69s/it]

Hình 4.19: Thời gian xử lý ảnh của mô hình

Biểu đồ minh họa quá trình giảm của hàm mất mát SSIM trong suốt 50 epochs huấn luyện mô hình Autoencoder. Có thể thấy cả train loss và test loss đều giảm dần một cách ổn định, phản ánh quá trình học tập hiệu quả của mô hình khi nắm bắt được các đặc trưng quan trọng từ dữ liệu mà không xuất hiện hiện tượng dao động bất thường hay bất ổn định trong quá trình tối ưu hóa.

Đường train loss luôn nằm dưới đường test loss, phù hợp với kỳ vọng trong huấn luyện mô hình học sâu. Khoảng cách giữa hai đường nhỏ và duy trì ổn định trong suốt quá trình huấn luyện, điều này khẳng định mô hình không bị overfitting và vẫn giữ được khả năng tổng quát hóa tốt khi đánh giá trên tập kiểm thử.

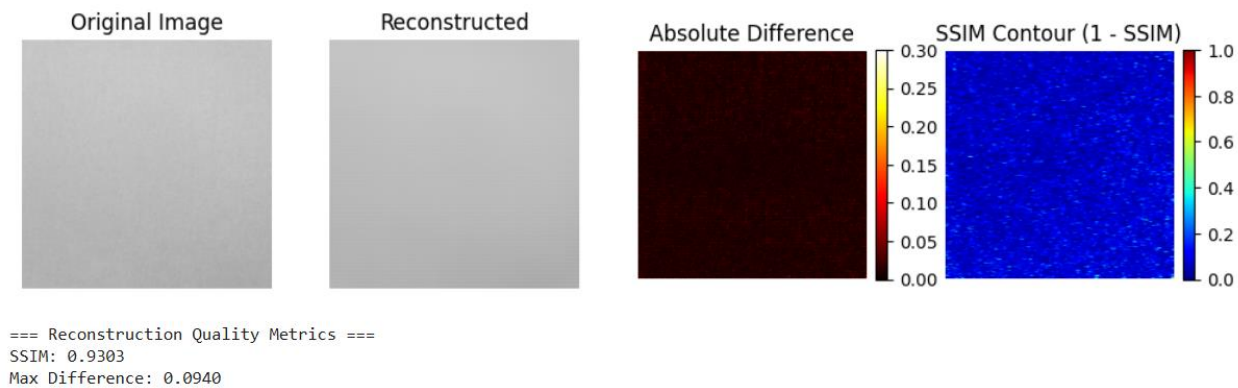
Trong giai đoạn đầu (khoảng 5 - 10 epochs đầu tiên), loss giảm nhanh chóng, thể hiện quá trình học các đặc trưng cơ bản của dữ liệu. Sau đó, từ epoch 10 trở đi, tốc độ giảm chậm lại đáng kể, cho thấy mô hình đang dần tiến tới điểm hội tụ và tập trung vào tối ưu hóa các chi tiết hơn. Đó là diễn hình của quá trình huấn luyện Autoencoder khi sử dụng bộ tối ưu AdamW, kết hợp với tốc độ học được điều chỉnh linh hoạt.

Đặc biệt, từ epoch 40 trở đi, cả hai đường loss đều tiệm cận một ngưỡng ổn định, với độ dốc gần như bằng không. Điều này cho thấy mô hình đã đạt được hiệu suất tối ưu với kiến trúc hiện tại và việc tiếp tục huấn luyện thêm không mang lại cải thiện đáng kể.

Nhìn chung, biểu đồ cho thấy mô hình Autoencoder đã được huấn luyện hiệu quả — không chỉ học tốt trên tập huấn luyện mà còn giữ được khả năng tổng quát hóa ấn tượng trên tập kiểm thử. Tuy nhiên, thời gian xử lý trung bình khoảng 4.69 giây/ảnh vẫn chưa đủ nhanh, do đó mô hình cần được tối ưu thêm để đáp ứng yêu cầu thực tế trong môi trường sản xuất.

4.9 Đánh giá mô hình

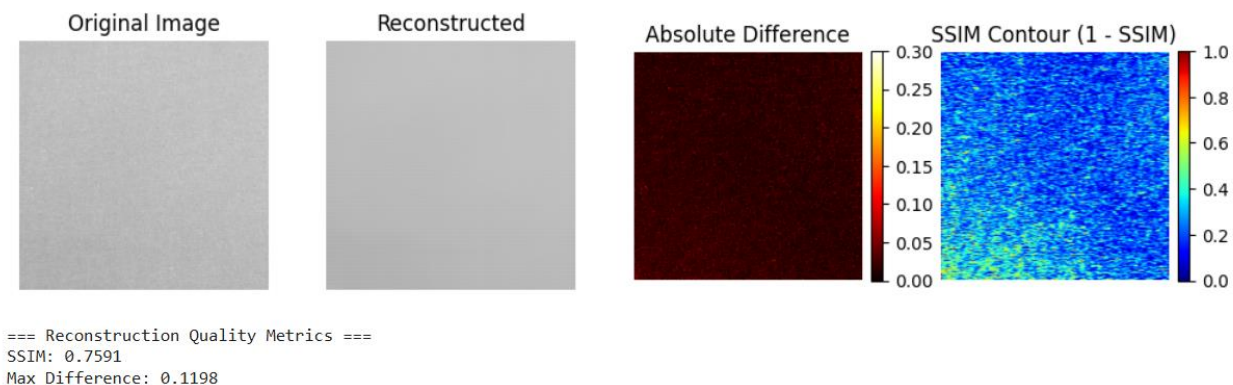
4.9.1 Đánh giá chất lượng tái tạo



Hình 4.20: Kiểm tra chất lượng tái tạo

- SSIM cao (~ 0.93): Chỉ số này thể hiện mức độ tái tạo rất tốt. So với các ảnh không lỗi, đây là mức SSIM lý tưởng, cho thấy mô hình khớp gần như hoàn toàn với ảnh gốc.
- Max Difference = 0.0940: Là mức sai khác rất thấp, chưa tới 0.1. Mức độ này thường gặp ở ảnh hoàn toàn không lỗi.
- Absolute Difference và SSIM Contour: Hầu như toàn màu tối và xanh dương - tức sai số nhỏ, đồng đều, không có vùng bất thường nổi bật.

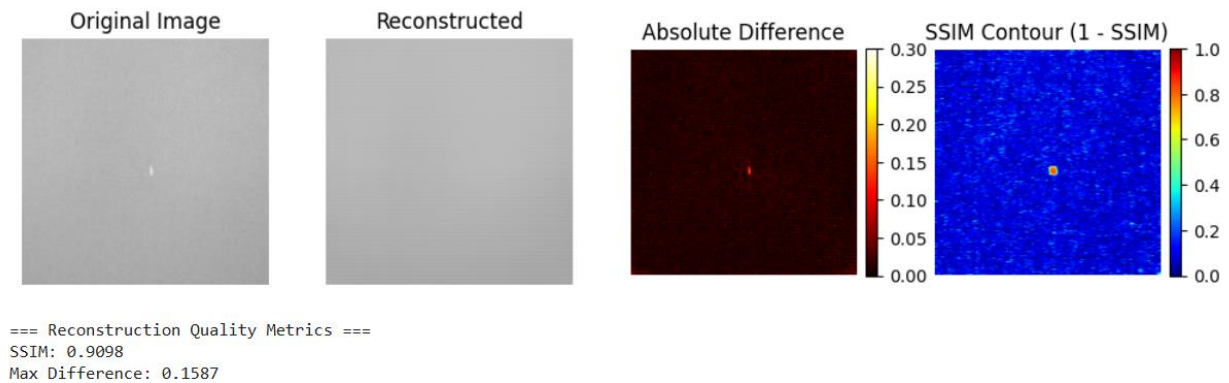
Kết luận: Ảnh không lỗi, mô hình hoạt động rất tốt.



Hình 4.21: Kiểm tra chất lượng tái tạo

- SSIM thấp (~ 0.76): So với ảnh không lỗi trước đó (SSIM ~ 0.90), mức này tương đối thấp, cho thấy mô hình không tái tạo chính xác hoàn toàn, dù hình ảnh đầu vào nhìn không có lỗi.

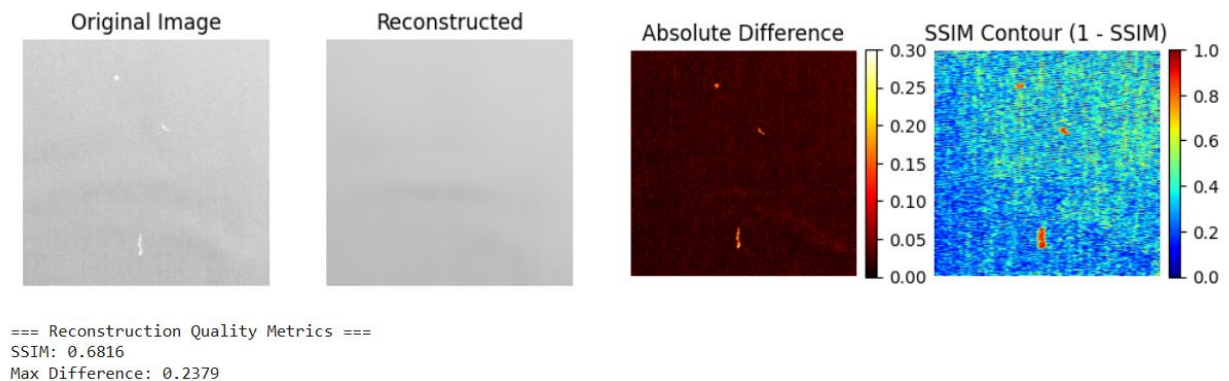
- Max Difference = 0.1108: Đây là mức sai khác nhỏ nhưng lại cao hơn so với các mẫu không lỗi điển hình (thường dưới 0.10).
- Absolute Difference và SSIM Contour: Phân bố sai số trải đều, không tập trung cục bộ, không có điểm nóng rõ ràng. Cho thấy đây có thể là lỗi nền hoặc nhiễu nhẹ do kết cấu vải, chứ không phải lỗi nghiêm trọng.
- **Kết luận:** Ảnh có dấu hiệu sai lệch nhẹ, nhưng chưa đủ rõ ràng để kết luận là lỗi. Nên được gắn nhãn là "ngghi ngờ" để kiểm thủ công.



Hình 4.22: Kiểm tra chất lượng tái tạo

- SSIM vẫn cao (~ 0.91): Tái tạo nhìn chung ổn định, phù hợp với đa số vùng ảnh.
- Max Difference = 0.1587: Giá trị tương đối thấp đến trung bình. Sự sai lệch tập trung vào một điểm nhỏ, cục bộ, không lan rộng ra toàn ảnh.
- Absolute Difference và SSIM Contour: Xuất hiện điểm "nóng" rõ ràng ở trung tâm, biểu hiện sai lệch cục bộ mạnh, đặc trưng cho các lỗi chấm nhỏ hoặc tạp chất.

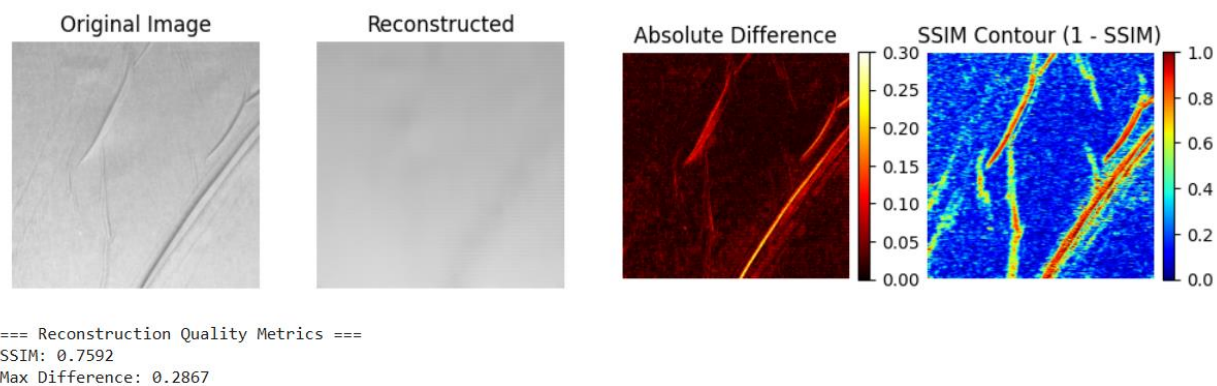
Kết luận: Ảnh có lỗi nhẹ, mô hình phát hiện chính xác bất thường cục bộ.



Hình 4.23: Kiểm tra chất lượng tái tạo

- SSIM thấp (~ 0.68): Đây là mức SSIM rất thấp, cho thấy mô hình tái tạo kém toàn cục - biểu hiện rõ rệt lỗi lan rộng hoặc nền bị biến đổi mạnh.
- Max Difference = 0.2379: Giá trị tương đối cao trong phạm vi $[0, 1]$ của độ lệch tái tạo. Dù chưa có ngưỡng cố định, so sánh tương quan với các mẫu khác cho thấy đây là một mức sai khác đáng chú ý, đặc biệt khi phần lớn vùng ảnh có độ lệch thấp
- Absolute Difference và SSIM Contour: Nhiều điểm đỏ, vàng xuất hiện rải rác và rõ ràng, đặc biệt có vùng sai số kéo dài - cho thấy lỗi vật lý hiện diện thực sự trên vải.

Kết luận: Ảnh có lỗi rõ rệt, dễ dàng nhận biết bằng cả mô hình và mắt thường.



Hình 4.24: Kiểm tra chất lượng tái tạo

- SSIM thấp (~ 0.76): Tái tạo không tốt, so với các ảnh không lỗi, đây là dấu hiệu rõ ràng của ảnh bị lỗi.
- Max Difference = 0.2867: Mức độ chênh lệch cao. Trong trường hợp chưa có ngưỡng lỗi xác định, giá trị này vẫn cho thấy mức sai khác đáng kể, đủ để nghi ngờ ảnh chứa lỗi.
- Absolute Difference và SSIM Contour: Có các dải màu nóng rõ ràng kéo dài theo chiều nếp gấp - mô hình không thể tái tạo được cấu trúc tuyến tính này.

Kết luận: Ảnh có lỗi nặng, độ lệch tái tạo lớn và dễ phân biệt.

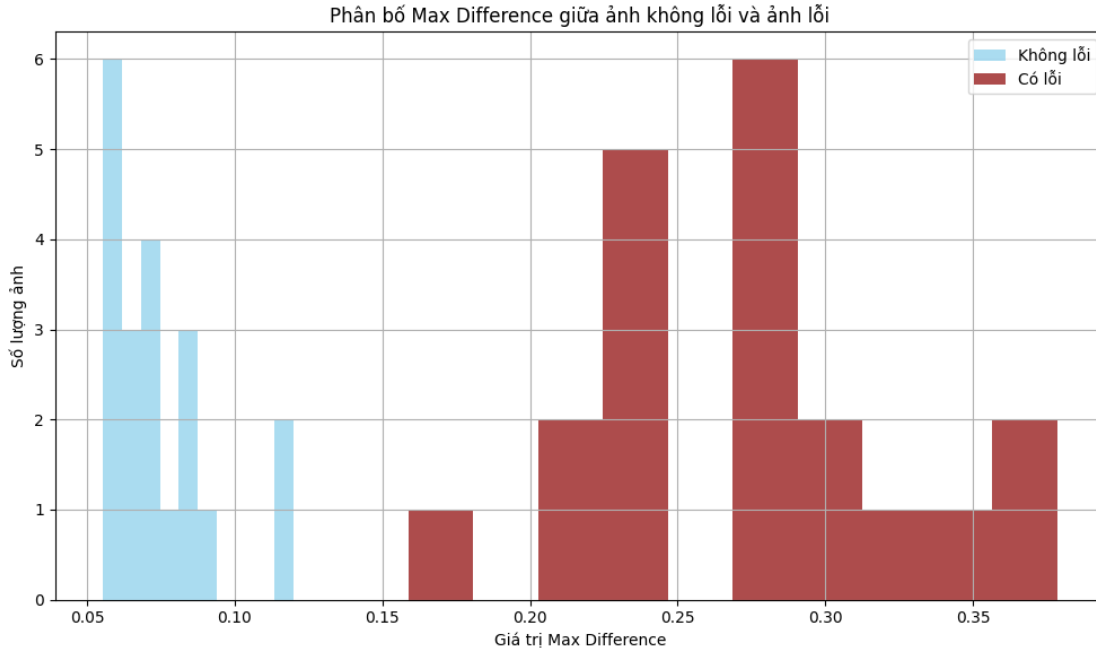
4.9.2 Chọn ngưỡng Max Difference cho mô hình phát hiện lỗi

Sau khi thử nghiệm trên tập dữ liệu bao gồm ảnh vải không lỗi và có lỗi, được đánh giá bằng mô hình Autoencoder. Mỗi ảnh được tính toán chỉ số Max Difference - thể hiện mức sai lệch lớn nhất giữa ảnh gốc và ảnh tái tạo. Thu thập và tổng hợp, thu được hai nhóm phân bố rõ ràng:

- Nhóm không lỗi: Max Difference $\in [0.05, 0.12]$ (Phụ lục 1)

- Nhóm có lỗi: Max Difference $\in [0.16, 0.38]$ (Phụ lục 2)

Biểu đồ bên dưới cho thấy hai cụm giá trị phân bố không chồng lấn và có một khoảng cách tách biệt tự nhiên giữa hai nhóm:



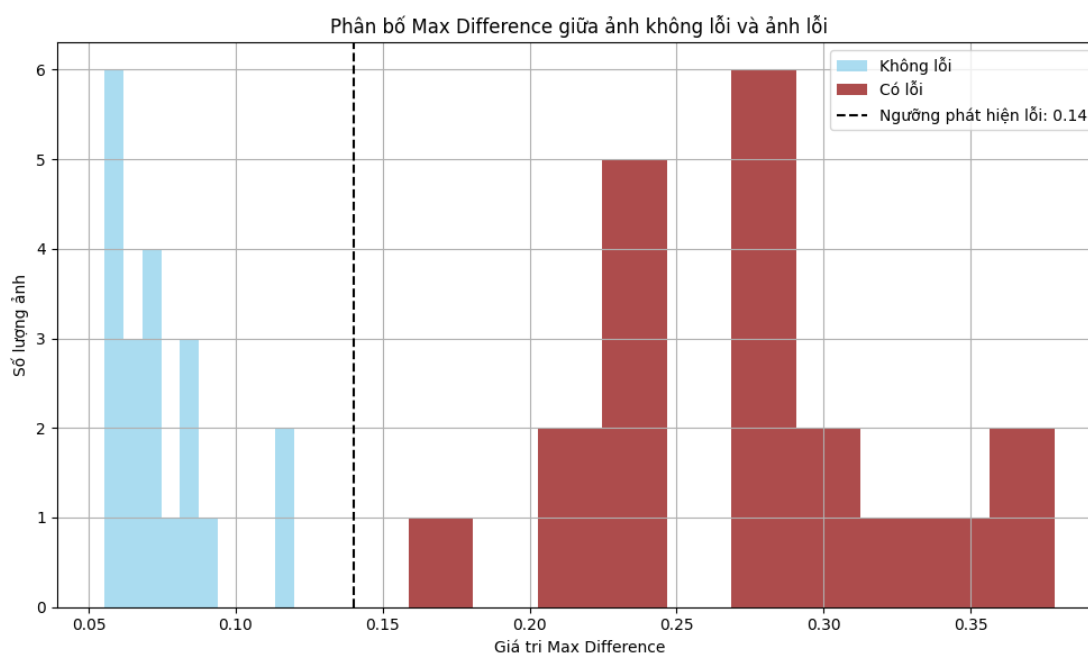
Hình 4.25: Phân bố Max Difference giữa ảnh lỗi và ảnh không lỗi

Dựa trên nguyên tắc phát hiện bất thường trong các mô hình học không giám sát, cụ thể là theo hướng tiếp cận mô hình DAGMM (Zong et al., 2018), việc đặt ngưỡng phát hiện lỗi tại khoảng giữa hai cụm phân bố đầu ra là cách làm phổ biến và hiệu quả trong các phương pháp phát hiện bất thường không giám sát. Theo (Zong et al., 2018), sự khác biệt giữa mẫu bình thường và bất thường thể hiện rõ qua sai số tái tạo và vị trí của mẫu trong không gian đặc trưng giảm chiều [27].

Vì vậy, lựa chọn ngưỡng tại vùng không chồng lấn giữa hai cụm không chỉ giúp tối ưu hoá khả năng phân loại, mà còn có tác dụng giảm thiểu sai số loại I (false positive) - báo nhầm lỗi cho ảnh tốt và sai số loại II (false negative) - bỏ sót lỗi thực sự.

Với những phân tích trên, Max Difference = 0.14 được đề xuất là lựa chọn phù hợp cho ngưỡng phân biệt lỗi của mô hình vì có đủ 2 yếu tố:

- Nằm trọn ngoài vùng an toàn của nhóm không lỗi
- Đủ nhạy để phát hiện các lỗi nhẹ nằm đầu cụm lỗi



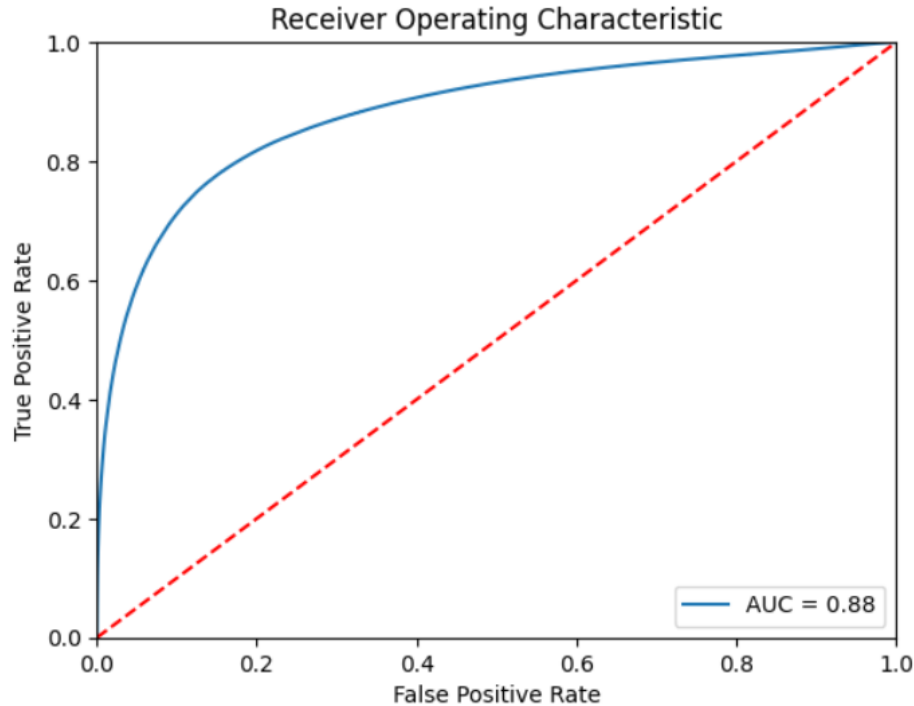
Hình 4.26: Phân bố Max Difference giữa ảnh lỗi và ảnh không lỗi

Bảng 4.4: Ngưỡng phân loại lỗi theo Max Difference

Phân loại	Điều kiện
Không lỗi	$\text{Max Difference} \leq 0.12$
Cảnh báo/ngghi ngờ	$0.12 < \text{Max Difference} \leq 0.14$
Lỗi nhẹ	$0.14 < \text{Max Difference} \leq 0.20$
Lỗi rõ/nặng	$\text{Max Difference} > 0.20$

4.9.3 Đánh giá độ nhạy của mô hình

AUC score on the test dataset: 0.8834186230703914

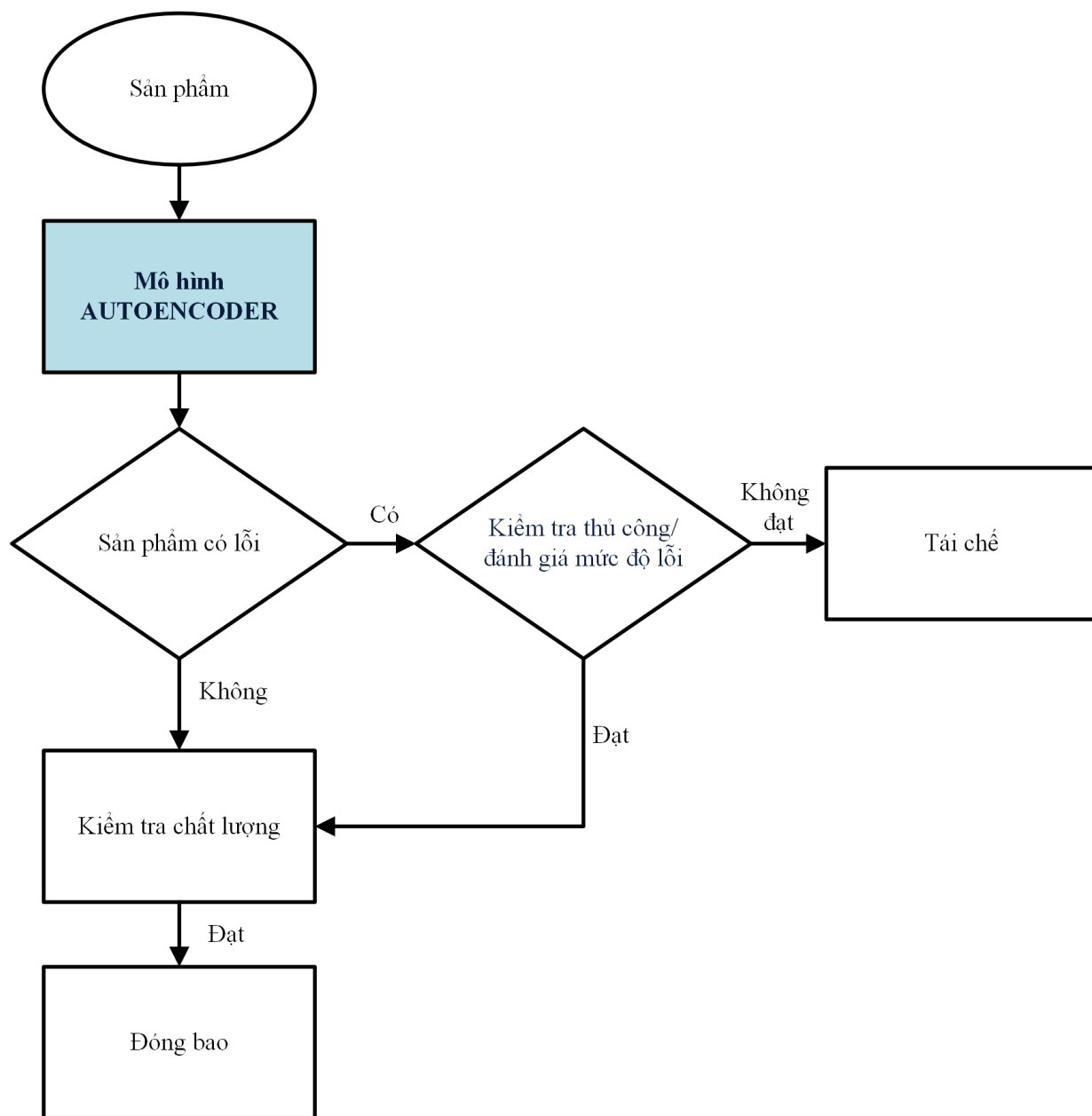


Hình 4.27: Hiệu suất phân loại của mô hình

Autoencoder được huấn luyện chủ yếu trên ảnh vải không lỗi, với mục tiêu học biểu diễn đặc trưng và tái tạo ảnh đầu vào. Khi mô hình gặp ảnh bị lỗi trong giai đoạn kiểm tra, sai số tái tạo (reconstruction error) thường cao hơn rõ rệt so với ảnh bình thường. Dựa vào các mức sai lệch giữa ảnh gốc và ảnh tái tạo, ta có thể vẽ ra một đường cong ROC bằng cách thay đổi ngưỡng để quyết định ảnh nào là lỗi. Với mỗi ngưỡng, ta tính xem mô hình nhận đúng bao nhiêu ảnh lỗi (độ nhạy) và nhận đúng bao nhiêu ảnh không lỗi (độ đặc hiệu). AUC khi đó là số liệu tóm tắt thể hiện năng lực tổng thể của mô hình trong việc phát hiện bất thường trên vải. Giá trị AUC cao (gần 1.0) cho thấy mô hình có khả năng nhận diện lỗi hiệu quả, giá trị gần 0.5 phản ánh mô hình không phân biệt được lỗi và không lỗi [10]. Việc sử dụng AUC trong bối cảnh này giúp đánh giá mô hình một cách khách quan, không phụ thuộc vào lựa chọn ngưỡng cụ thể và đặc biệt phù hợp khi dữ liệu lỗi có số lượng hạn chế hoặc phân bố không cân bằng.

Giá trị AUC (Area Under the Curve) được tính trên tập kiểm tra là 0.88, cho thấy mô hình có khả năng phân biệt tốt giữa hai lớp.

CHƯƠNG 5: ĐỀ XUẤT PHƯƠNG ÁN CẢI TIẾN



Hình 5.1: Sơ đồ phương án cải tiến quy trình kiểm tra chất lượng tại xưởng thành phẩm
Trong bối cảnh sản xuất công nghiệp hiện đại, đặc biệt là ngành dệt may, việc kiểm soát chất lượng nguyên liệu đầu vào, cụ thể là vải đóng vai trò then chốt trong việc đảm bảo độ

ổn định của sản phẩm thành phẩm, giảm thiểu chi phí phát sinh và gia tăng độ tin cậy của doanh nghiệp trong chuỗi cung ứng.

Xuất phát từ yêu cầu thực tiễn tại xưởng thành phẩm vừa công ty, nghiên cứu đề xuất một phương án cải tiến quy trình kiểm vải bằng cách tích hợp mô hình Autoencoder nhằm hỗ trợ phát hiện lỗi tự động trên bề mặt vải. Phương án đề xuất sử dụng camera công nghiệp để quét ảnh bề mặt vải, sau đó đưa ảnh vào mô hình Autoencoder đã được huấn luyện trên tập dữ liệu vải không lỗi. Nhờ khả năng học đặc trưng và tái tạo hình ảnh, mô hình sẽ phát hiện các vùng bất thường - chẳng hạn như lẫn tạp chất, lỗi dệt, lỗ thủng - thông qua sai số tái tạo cao tại các khu vực không khớp với dữ liệu học trước đó.

Các cuộn vải không lỗi sẽ được chuyển thẳng sang công đoạn kiểm tra chất lượng, sau đó chuyển đến đóng gói, trong khi đó các cuộn có lỗi sẽ được phân luồng để kiểm tra lại bằng mắt thường và xử lý. Điều này cho phép tập trung nguồn lực con người vào đúng vị trí cần kiểm tra, thay vì phải kiểm toàn bộ lô vải như quy trình truyền thống.

Phương án này giúp giảm đáng kể khối lượng kiểm tra thủ công, góp phần giảm mỏi mắt cho công nhân, hạn chế nguy cơ bỏ sót lỗi nhỏ và nâng cao độ chính xác trong công đoạn đánh giá chất lượng. Ngoài ra, quá trình xử lý ảnh bằng mô hình diễn ra nhanh chóng, góp phần rút ngắn thời gian kiểm hàng, giảm độ trễ trong chu trình sản xuất và tiết kiệm chi phí vận hành.

CHƯƠNG 6: ĐÁNH GIÁ VÀ HƯỚNG PHÁT TRIỂN TRONG TƯƠNG LAI

6.1 Đánh giá tính khả thi

6.1.1 Tính khả thi về kỹ thuật

Việc ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) vào khâu phát hiện khuyết tật sản phẩm vải là một giải pháp kỹ thuật có tính thực tiễn cao trong môi trường doanh nghiệp. Autoencoder là mô hình học sâu đặc biệt phù hợp với bài toán phát hiện các mẫu bất thường trên ảnh - là các khuyết tật xuất hiện ngẫu nhiên, không có hình dạng cố định như tạp chất, lỗ thủng, đứt sợi, sợi thô...

Từ góc độ doanh nghiệp, hệ thống này có thể hoạt động tự động, liên tục và chính xác, giảm phụ thuộc vào nhân công thủ công - vốn dễ mắc sai sót, mệt mỏi hoặc thiếu đồng nhất giữa các ca làm việc.

Về mặt triển khai, mô hình có thể được tích hợp vào hệ thống hiện tại của xưởng (hệ thống camera trên dây chuyền, các module kiểm tra vải...) mà không yêu cầu thay đổi lớn về cơ sở hạ tầng.

Ngoài ra, hệ thống có thể được huấn luyện riêng cho từng dòng sản phẩm hoặc từng loại vải, giúp tăng độ chính xác và khả năng tùy biến linh hoạt theo yêu cầu sản xuất.

6.1.2 Tính khả thi về thời gian triển khai

Trong môi trường doanh nghiệp, việc triển khai một hệ thống kiểm định tự động bằng Autoencoder có thể thực hiện theo các giai đoạn rõ ràng:

Bảng 6.1: Kế hoạch triển khai hệ thống

Giai đoạn	Công việc cần làm	Thời gian dự kiến
Khảo sát	Khảo sát, thu thập ảnh mẫu từ các lô sản phẩm tiêu chuẩn và các lô lỗi thực tế	2 - 3 tuần
Huấn luyện	Huấn luyện mô hình dựa trên dữ liệu nội bộ, tùy chỉnh hệ thống nhận dạng phù hợp với điều kiện ánh sáng, loại vải và vị trí camera tại xưởng	3 - 4 tuần
Thử nghiệm	Triển khai thử nghiệm tại một khu vực sản xuất cụ thể, so sánh kết quả kiểm tra tự động với nhân công	4 - 6 tuần
Mở rộng	Đánh giá, tối ưu và mở rộng áp dụng toàn bộ xưởng nếu kết quả đạt yêu cầu	2 tuần

Kế hoạch triển khai có tổng thời gian dự kiến từ 3 đến 4 tháng, tương đối phù hợp với chu kỳ đầu tư công nghệ ngắn hạn trong doanh nghiệp. Việc chia thành bốn giai đoạn - gồm khảo sát, huấn luyện mô hình, thử nghiệm thực tế và mở rộng - thể hiện một lộ trình có hệ thống, thường được áp dụng trong các dự án công nghệ nhằm giảm thiểu rủi ro khi thay đổi quy trình sản xuất hiện hành.

Về mặt tích cực, kế hoạch cho phép thử nghiệm mô hình trực tiếp trong môi trường sản xuất, đồng thời so sánh với kết quả kiểm tra thủ công. Đây là cách tiếp cận thực tiễn, giúp doanh nghiệp đánh giá cụ thể hiệu quả của hệ thống về độ chính xác, tốc độ phát hiện lỗi và mức độ tương thích với quy trình vận hành hiện tại. Tuy nhiên, nếu hệ thống ảnh hưởng đến tốc độ dây chuyền hoặc yêu cầu điều chỉnh quy trình quá nhiều, khả năng chấp nhận từ phía doanh nghiệp sẽ bị hạn chế.

Về tổng thể, tính khả thi của kế hoạch ở mức trung bình đến cao, tùy thuộc vào mức độ sẵn sàng của doanh nghiệp về ba yếu tố chính: hạ tầng kỹ thuật, chất lượng dữ liệu đầu vào và khả năng phối hợp giữa nhóm triển khai và bộ phận sản xuất. Với các doanh nghiệp chưa từng ứng dụng công nghệ thị giác máy tính, cần đặc biệt lưu ý đến thời gian tinh chỉnh mô hình, đào tạo nội bộ và chi phí phát sinh trong quá trình tích hợp. Khi những điều kiện nền tảng này được đảm bảo, kế hoạch có thể triển khai hiệu quả trên quy mô thí điểm, làm tiền đề cho việc mở rộng sau này.

6.1.3 Tính khả thi về nguồn lực

Về nhân sự, doanh nghiệp không cần tuyển mới chuyên gia AI từ đầu mà có thể hợp tác với đơn vị triển khai công nghệ bên ngoài (startup, nhóm nghiên cứu hoặc thuê dịch vụ tích hợp AI). Sau khi cài đặt, hệ thống có thể vận hành với sự giám sát của kỹ thuật viên hiện có, chỉ cần đào tạo thêm ở mức cơ bản.

Về thiết bị, doanh nghiệp bổ sung hệ thống camera kiểm tra chất lượng có độ phân giải cao, phần mềm xử lý ảnh và máy tính có GPU. Nếu đầu tư thêm phần cứng, chi phí không quá lớn so với hiệu quả mang lại (máy tính GPU, server nhỏ hoặc thiết bị Edge AI có giá vừa phải).

Về chi phí phần mềm, hệ thống có thể sử dụng các công cụ mã nguồn mở, do đó chi phí giấy phép gần như bằng 0. Chi phí chủ yếu là ở công đoạn triển khai ban đầu, đào tạo mô hình và hiệu chỉnh theo điều kiện thực tế của nhà máy.

6.1.4 Khả năng mở rộng và ứng dụng

Một trong những ưu điểm lớn của hệ thống AI dùng Autoencoder là khả năng mở rộng linh hoạt:

- Có thể áp dụng trên nhiều dòng sản phẩm khác nhau chỉ bằng cách huấn luyện lại mô hình với dữ liệu tương ứng.
- Có thể tích hợp với băng chuyền, hoặc hệ thống ERP để tạo ra chuỗi kiểm tra - ghi nhận - báo lỗi - phản hồi hoàn chỉnh.
- Có thể phát triển thêm tính năng phân loại lỗi, thống kê tỷ lệ lỗi theo ca/khu vực/máy móc, từ đó giúp nhà quản lý nhanh chóng nhận diện vấn đề và cải tiến quy trình.

6.1.5 Hạn chế và rủi ro

Mặc dù khả thi, một số rủi ro doanh nghiệp cần tính đến gồm:

Bảng 6.2: Hạn chế và rủi ro khi triển khai mô hình

Rủi ro	Hạn chế
Chất lượng dữ liệu ảnh đầu vào	Nếu ánh sáng, độ nét hoặc vị trí camera không ổn định, mô hình sẽ dễ nhầm lẫn. Do đó, cần có quy chuẩn ghi hình chặt chẽ.

Thiếu dữ liệu lỗi thực tế	Nhiều xưởng có tỷ lệ lỗi thấp, dẫn đến thiếu dữ liệu để mô hình học cách phát hiện lỗi. Doanh nghiệp cần lưu trữ và phân loại dữ liệu lâu dài để huấn luyện chính xác.
Chi phí triển khai ban đầu	Dù không quá cao, doanh nghiệp vẫn cần cam kết đầu tư ban đầu, nhất là về kỹ thuật và tích hợp hệ thống.

6.2 Hướng phát triển trong tương lai

6.2.1 Tối ưu hiệu suất xử lý của mô hình

Tối ưu hiệu suất xử lý của mô hình nhằm đáp ứng tốt hơn yêu cầu thời gian thực trong môi trường sản xuất. Cụ thể, có thể nghiên cứu áp dụng các kỹ thuật như rút gọn kiến trúc mô hình để giảm độ trễ, triển khai trên GPU chuyên dụng thay vì CPU thông thường hoặc ứng dụng các giải pháp inference tốc độ cao giúp rút ngắn thời gian xử lý mỗi ảnh xuống mức phù hợp với thực tế vận hành dây chuyền.

6.2.2 Nâng cấp mô hình để phân loại lỗi chi tiết

Thay vì chỉ dừng lại ở mức phát hiện “có lỗi hay không”, mô hình có thể được kết hợp với các kỹ thuật học sâu khác (CNN, Transfer Learning...) để phân loại cụ thể loại lỗi như: Lỗi tạp chất, lỗi kết đầu, lỗi sợi thô, lỗi đứt sợi...

Việc phân loại chi tiết giúp bộ phận kiểm định và quản lý chất lượng đưa ra phản ứng phù hợp, nhanh chóng xác định nguyên nhân và xử lý sự cố.

Từ đó có thể phân tích dữ liệu lỗi để hỗ trợ quản lý sản xuất:

Từ dữ liệu lỗi được hệ thống tự động ghi nhận và lưu trữ trong quá trình kiểm tra, doanh nghiệp có thể khai thác để phục vụ công tác phân tích, giám sát và cải tiến quy trình sản xuất một cách chủ động và có cơ sở khoa học.

Bảng 6.3: Phân tích dữ liệu lỗi

Thông tin	Loại lỗi				
	Đứt sợi	Tạp chất	Kết đầu	Sợi dọc thô	Lỗ thủng
Ca làm việc					
Loại vải					
Số lượng					
Máy móc liên quan					
Nhân sự vận hành					

Hỗ trợ doanh nghiệp phát hiện điểm yếu trong quy trình sản xuất

Thông qua thống kê và phân tích dữ liệu lỗi theo thời gian, theo mã ca làm việc, loại máy móc, nhân sự vận hành hoặc loại vải, doanh nghiệp có thể nhanh chóng xác định được những nút thắt trong dây chuyền sản xuất - nơi có tần suất lỗi cao bất thường. Từ đó xem xét nguyên nhân và đưa ra giải pháp.

Bảng 6.4: Vấn đề và nghi vấn khi phân tích dữ liệu

Vấn đề	Nghi vấn
Một loại máy may cụ thể có tỷ lệ sản phẩm lỗi cao hơn hẳn các máy còn lại.	Nghi vấn về lỗi thiết bị hoặc cần hiệu chỉnh lại máy.
Lỗi xảy ra thường xuyên vào một khung giờ nhất định.	Có thể liên quan đến thay ca, thao tác của công nhân, hoặc yếu tố môi trường (ánh sáng, nhiệt độ).
Một mẫu vải mới có tỷ lệ phát hiện lỗi cao hơn hẳn các mẫu trước đó.	Cần xem xét chất lượng nguyên vật liệu hoặc quy trình tiền xử lý.

6.2.3 Mở rộng sang các khâu sản xuất khác

Một trong những hướng phát triển tiềm năng, có giá trị ứng dụng cao trong tương lai là việc mở rộng mô hình Autoencoder và công nghệ xử lý ảnh sang các công đoạn khác trong toàn bộ chuỗi sản xuất của doanh nghiệp dệt may. Điều này không chỉ dừng lại ở việc kiểm tra vải sau dệt - như mục tiêu ban đầu - mà còn có thể được tùy biến và huấn luyện lại để áp dụng hiệu quả vào các khâu sản xuất quan trọng khác như:

Kiểm tra sợi trước khi dệt: Phát hiện các lỗi như gãy sợi, sợi dơ, không đều màu hoặc không đúng thông số kỹ thuật trước khi đưa vào quá trình dệt. Đây là bước quan trọng giúp đảm bảo đầu vào ổn định cho chất lượng vải thành phẩm.

Kiểm tra sản phẩm sau may (quần áo thành phẩm): Ứng dụng công nghệ xử lý ảnh để phát hiện lỗi may lệch, đường chỉ hở, thừa chỉ, nhăn nếp không đúng chuẩn hoặc lỗi không đối xứng trong thiết kế. Việc kiểm tra tự động ở khâu này giúp bảo đảm thẩm mỹ và độ chính xác cao cho thành phẩm.

Kiểm tra công đoạn in, nhuộm: Giám sát và phát hiện lỗi in lệch, lem màu, không đều màu hoặc sai mẫu in. Đây là những lỗi rất khó phát hiện bằng mắt thường nếu không có hệ thống hỗ trợ chính xác và ổn định.

KẾT LUẬN

Đề tài đã hoàn thành mục tiêu xây dựng hệ thống kiểm tra lỗi ngoại quan vải tự động bằng mô hình học sâu Autoencoder theo hướng không giám sát. Thông qua việc huấn luyện trên dữ liệu vải không lỗi, mô hình có thể phát hiện các bất thường bằng sai số tái tạo mà không cần phụ thuộc vào dữ liệu lỗi - một lợi thế lớn trong môi trường sản xuất thực tế. Kết quả thử nghiệm cho thấy mô hình đạt $AUC = 0.88$, thể hiện khả năng phân biệt hiệu quả giữa sản phẩm lỗi và không lỗi, đồng thời đáp ứng tốt yêu cầu về tốc độ xử lý và độ ổn định trong môi trường công nghiệp. Tuy nhiên, mô hình đang chạy ở giai đoạn thử nghiệm trên nền tảng chung, chưa được tối ưu hóa hạ tầng phần cứng nên thời gian xử lý trung bình khoảng 4.69 giây/ảnh vẫn chưa đủ nhanh, do đó mô hình cần được tối ưu thêm để đáp ứng yêu cầu thực tế trong môi trường sản xuất.

Đóng góp nổi bật của đề tài là đề xuất và kiểm chứng khả năng ứng dụng thực tế của mô hình Autoencoder vào dây chuyền sản xuất ngành dệt may - cụ thể tại công đoạn kiểm tra chất lượng vải tại xưởng thành phẩm. Mô hình không chỉ giảm thiểu sai sót do con người gây ra mà còn góp phần tối ưu nhân lực, rút ngắn thời gian kiểm hàng và tăng hiệu suất kiểm tra.

Tuy nhiên, do kinh nghiệm nghiên cứu của em còn hạn chế và phạm vi thực hiện chỉ tập trung vào một khâu cụ thể trong quy trình sản xuất, nên kết quả của nghiên cứu chưa thể đại diện cho toàn doanh nghiệp. Các đề xuất, kiến nghị được trình bày trong luận văn vẫn mang tính định hướng và lý thuyết, cần được kiểm chứng và hiệu chỉnh thông qua thực tế triển khai, cũng như đón nhận thêm ý kiến đóng góp từ thầy cô và doanh nghiệp. Dù vậy, nghiên cứu kỳ vọng sẽ trở thành nền tảng để phát triển các đề tài tiếp theo, với phạm vi rộng hơn và chiều sâu ứng dụng cao hơn, hướng đến tự động hóa toàn diện trong quản lý chất lượng sản phẩm dệt may.

TÀI LIỆU THAM KHẢO

Tài liệu web tiếng anh:

[2] World Fashion Exchange. *The Importance of Quality Control in Garment Manufacturing*.

<https://www.worldfashionexchange.com/blog/the-importance-of-quality-control-in-garment-manufacturing/>

[3] ECI Solutions. *Top 10 Reasons for Quality Control in Manufacturing*.

<https://www.ecisolutions.com/blog/manufacturing/top-10-reasons-for-quality-control-manufacturing/>

[12] Wikipedia contributors. (2024, May 15). *Structural similarity*. Wikipedia.

https://en.wikipedia.org/wiki/Structural_similarity

Tài liệu web tiếng việt:

[9] Tiêu chuẩn HACCP Codex 2020 được Ủy ban Codex chính thức ban hành ngày 23/11/2020, mang số hiệu CXC 1-1969, mục 3.4.

<https://isoq.vn/tcvn/tieu-chuan-haccp-codex-2020/>

Bài báo cáo khoa học tiếng anh:

[1] Liu, J., Xie, G., Wang, J., Li, S., Wang, C., Zheng, F., & Jin, Y. (2023). *Deep industrial image anomaly detection: A survey*.

[4] Saad, M. (2022). *The Role of Total Quality Management in the Textile Industry*.

[5] Neyestani, B. (2017). *Seven Basic Tools of Quality Control: An Appropriate Tools for Solving Quality Problems in the Organizations*

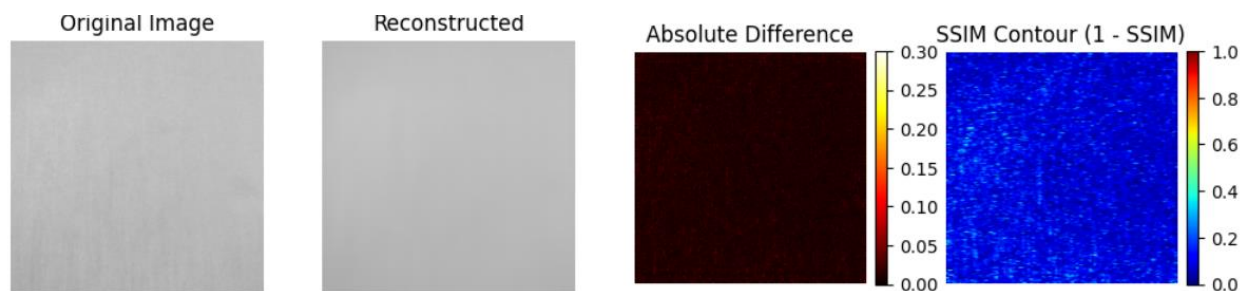
[6] Nicolae, R., Nedelcu, A., & Dumitrascu, A. E. (2015). *Improving the quality of industrial products by applying the Pareto chart*. Review of the Air Force Academy, (3), 169.

- [7] Bilsel, R. U., & Lin, D. K. J. (2012). *Ishikawa Cause and Effect Diagrams Using Capture Recapture Techniques*. *Quality Technology & Quantitative Management*, 9(2), 137–152.
- [8] Magar, V.M. and Shinde, V.B. (2014) *Application of 7 Quality Control (7 QC) Tools for Continuous Improvement of Manufacturing Processes*. *International Journal of Engineering Research and General Science*, 2, 364-371.
- [10] Antony, J., McDermott, O., & Sony, M. (2023). *Revisiting Ishikawa's Original Seven Basic Tools of Quality Control: A Global Study and Some New Insights*.
- [11] Ozek, A., Seckin, M., Demircioglu, P., & Boglekci, I. (2025). *Artificial intelligence driving innovation in textile defect detection*. *Textiles*, 5(2), 12.
- [13] Thakur, K. V., Damodare, O. H., & Sapkal, A. M. (2016). *Identification of suited quality metrics for natural and medical images*. *Signal & Image Processing: An International Journal*, 7(3), 29–43.
- [14] Berahmand, K., Daneshfar, F., Salehi, E. S., Li, Y., & Xu, Y. (2024). *Autoencoders and their applications in machine learning: A survey*.
- [15] Bergmann, P., Löwe, S., Fauser, M., Sattlegger, D., & Steger, C. (2018). *Improving Unsupervised Defect Segmentation by Applying Structural Similarity to Autoencoders*.
- [16] Pongetti, J., Kipouros, T., Emmanuelli, M., Ahlfeld, R., & Shahpar, S. (2021). *Using autoencoders and output consolidation to improve machine learning models for turbomachinery applications*. In *Proceedings of the ASME 2021 Turbomachinery Technical Conference & Exposition*.
- [17] Gholamy, A., Kreinovich, V., & Kosheleva, O. (n.d.). *Why 70/30 or 80/20 Relation Between Training and Testing Sets: A Pedagogical Explanation*. University of Texas at El Paso.
- [18] Tan, M. and Le, Q.V. (2019) *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*
- [19] Zhao, H., Gallo, O., Frosio, I., & Kautz, J. (2018). *Loss Functions for Image Restoration with Neural Networks*.

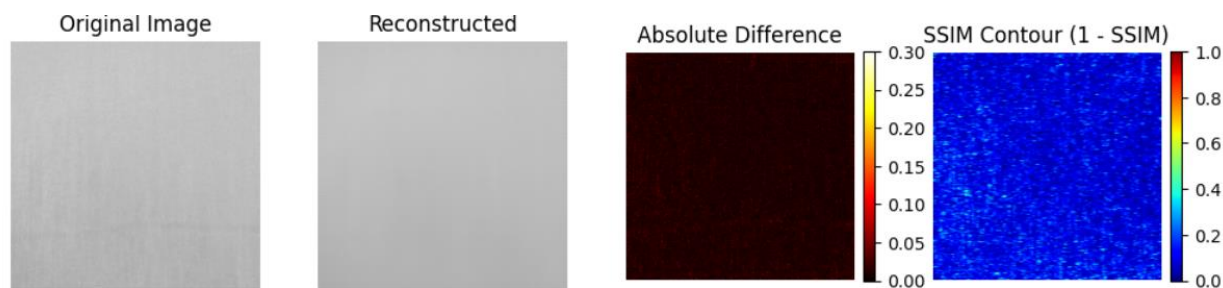
- [20] Seif, G., & Androutsos, D. (2018, April). *Edge-based loss function for single image super-resolution*. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 1882–1886)
- [21] Jiang, Y., & Li, J. (2020). *Generative adversarial network for image super-resolution combining texture loss*. Applied Sciences, 10(5), 1729.
- [22] He, X., & Cheng, J. (2022). *Revisiting L1 Loss in Super-Resolution: A Probabilistic View and Beyond*.
- [23] Torabi, H., Mirtaheri, S. L., & Greco, S. (2023). *Practical autoencoder based anomaly detection by using vector reconstruction error*. Cybersecurity, 6(1).
- [24] Wang, Z., Simoncelli, E. P., & Bovik, A. C. (2003). *Multiscale structural similarity for image quality assessment*. In Proceedings of the 37th Asilomar Conference on Signals, Systems & Computers (Vol. 2, pp. 1398–1402).
- [25] Liu, G., Reda, F. A., Shih, K. J., Wang, T.-C., Tao, A., & Catanzaro, B. (2018). *Image Inpainting for Irregular Holes Using Partial Convolutions*.
- [26] Dosovitskiy, A., & Brox, T. (2016). *Generating images with perceptual similarity metrics based on deep networks*. In Advances in Neural Information Processing Systems (NeurIPS 2016).
- [27] Zong, B., Song, Q., Min, M. R., Cheng, W., Lumezanu, C., Cho, D., & Chen, H. (2018). *Deep autoencoding gaussian mixture model for unsupervised anomaly detection*. In International Conference on Learning Representations (ICLR). NEC Laboratories America & Washington State University.

PHỤ LỤC

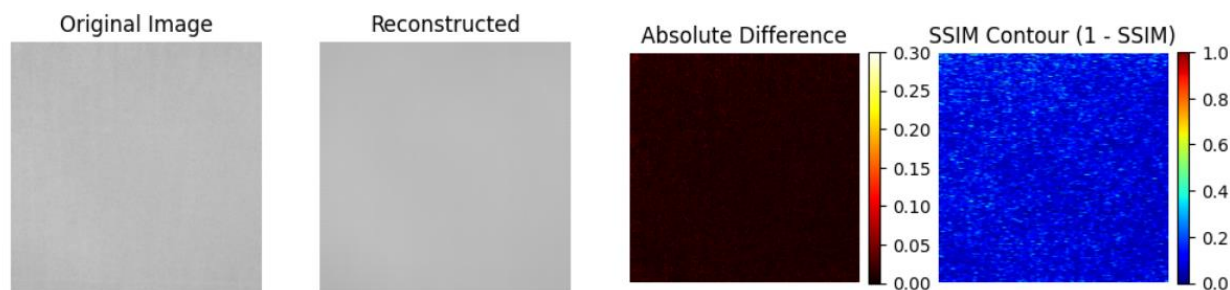
Phụ lục 1: Ảnh kết quả thu thập của nhóm không lỗi để chọn ngưỡng phân loại



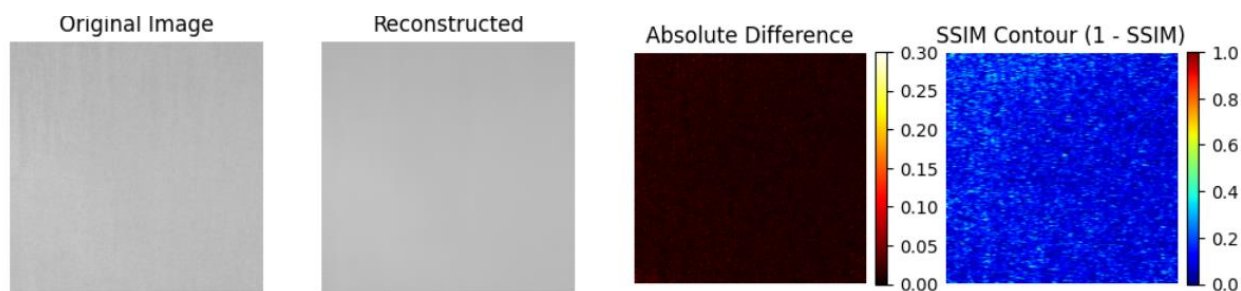
=== Reconstruction Quality Metrics ===
SSIM: 0.9099
Max Difference: 0.0790



=== Reconstruction Quality Metrics ===
SSIM: 0.9020
Max Difference: 0.0645

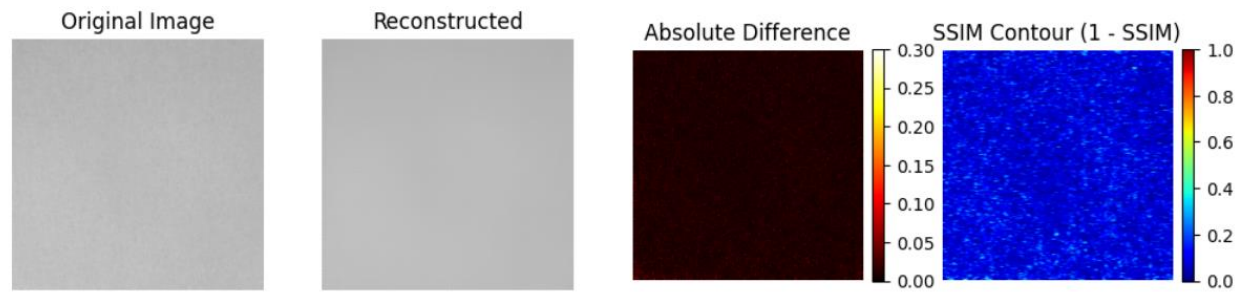


=== Reconstruction Quality Metrics ===
SSIM: 0.9048
Max Difference: 0.0596

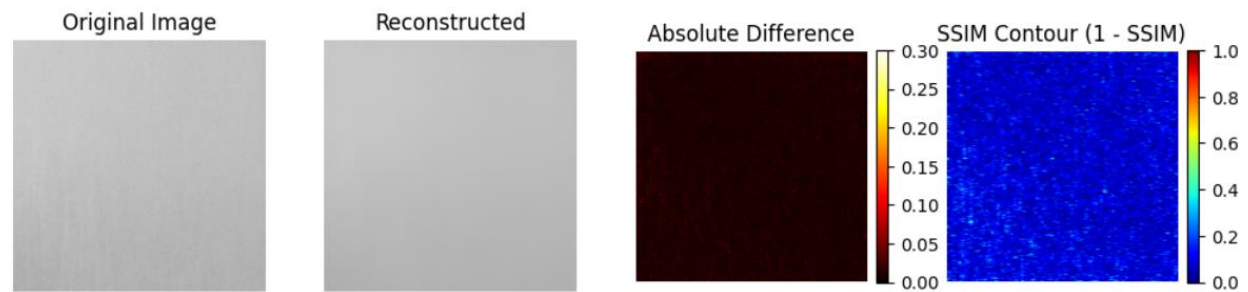


=== Reconstruction Quality Metrics ===
SSIM: 0.8864
Max Difference: 0.0905

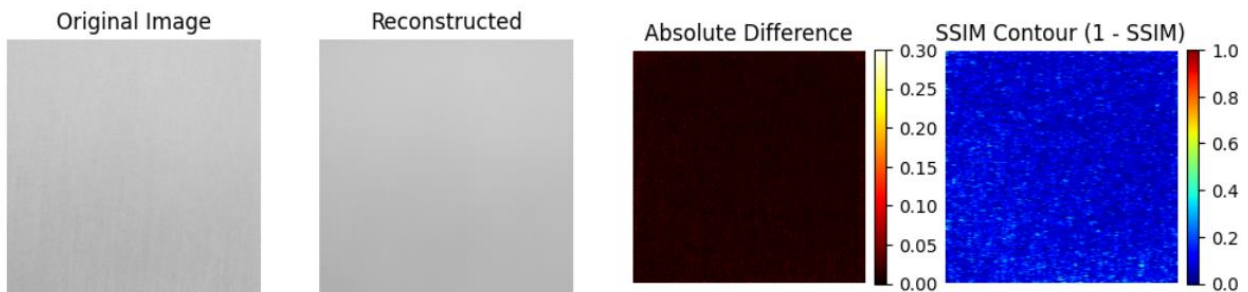
Đề tài: Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải



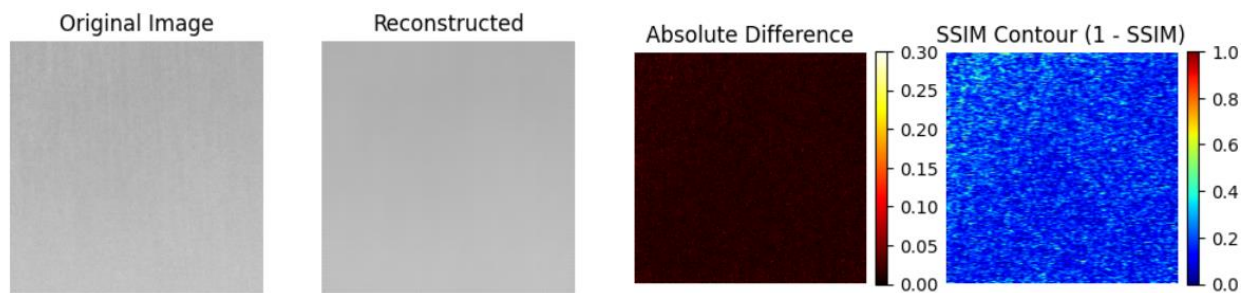
=== Reconstruction Quality Metrics ===
SSIM: 0.9115
Max Difference: 0.0713



=== Reconstruction Quality Metrics ===
SSIM: 0.9220
Max Difference: 0.0609

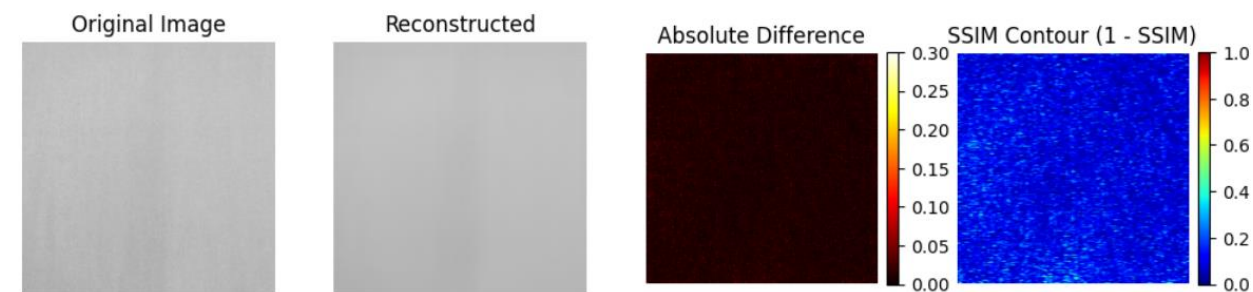


=== Reconstruction Quality Metrics ===
SSIM: 0.9154
Max Difference: 0.0605

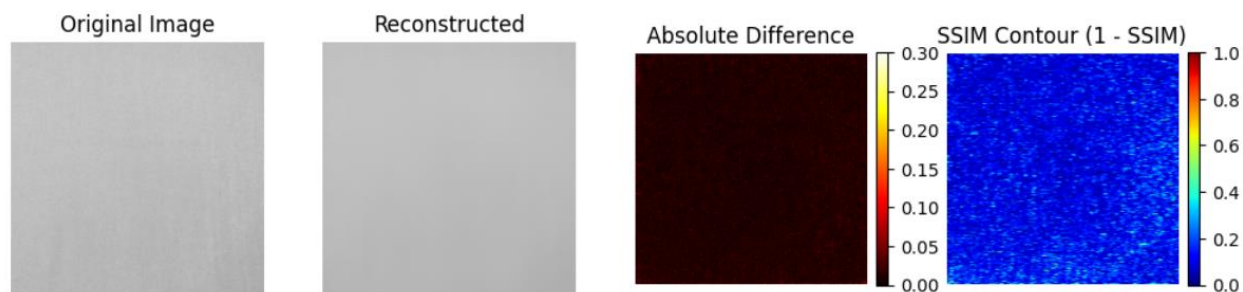


=== Reconstruction Quality Metrics ===
SSIM: 0.8394
Max Difference: 0.0843

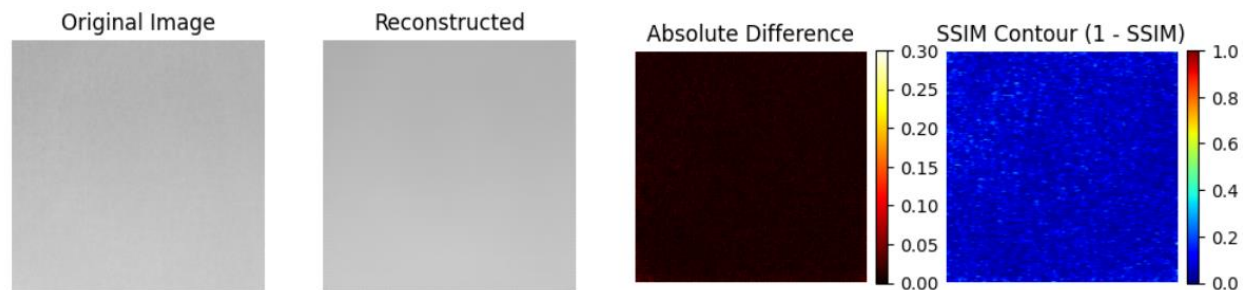
Đề tài: Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải



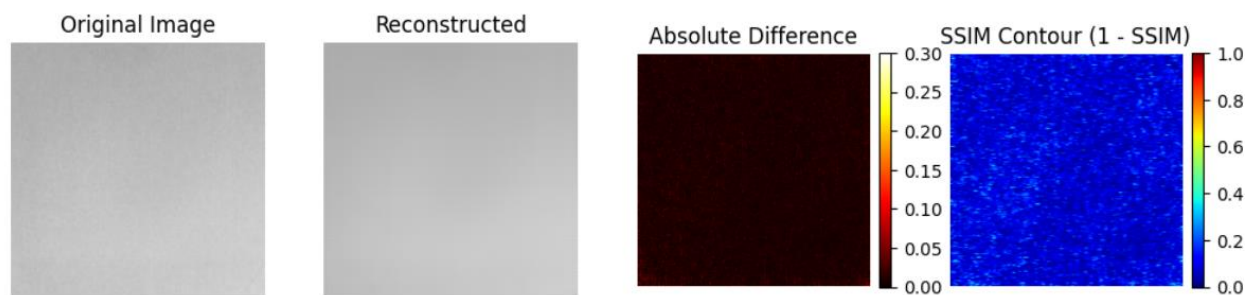
=== Reconstruction Quality Metrics ===
SSIM: 0.9000
Max Difference: 0.0660



=== Reconstruction Quality Metrics ===
SSIM: 0.8879
Max Difference: 0.0862

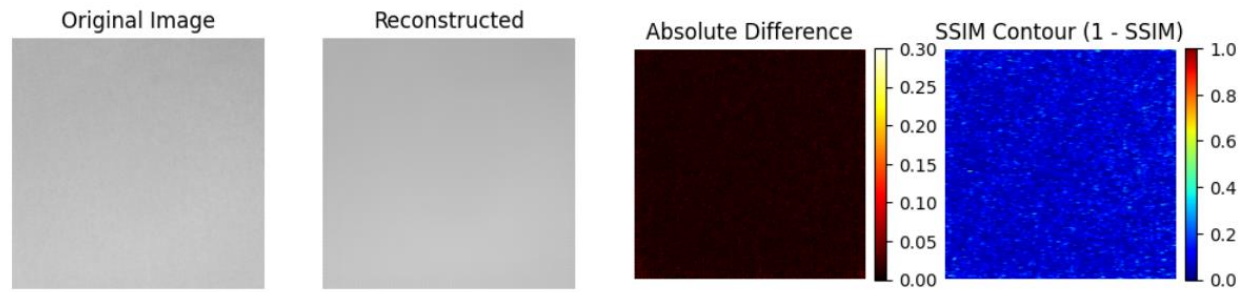


=== Reconstruction Quality Metrics ===
SSIM: 0.9370
Max Difference: 0.0553

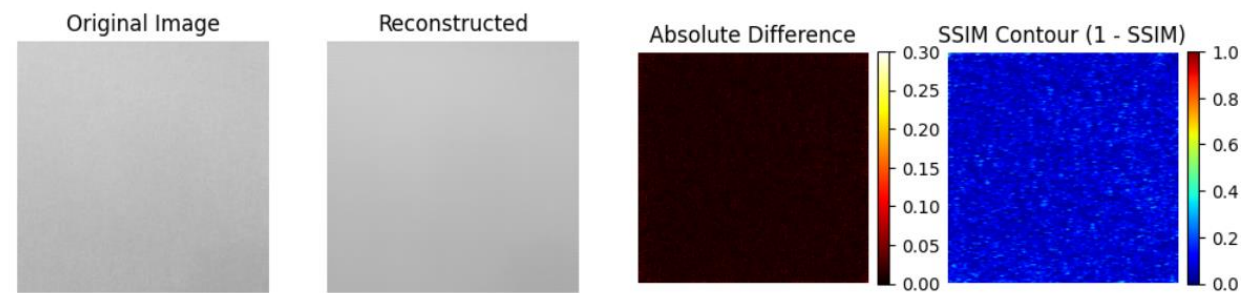


=== Reconstruction Quality Metrics ===
SSIM: 0.9165
Max Difference: 0.0687

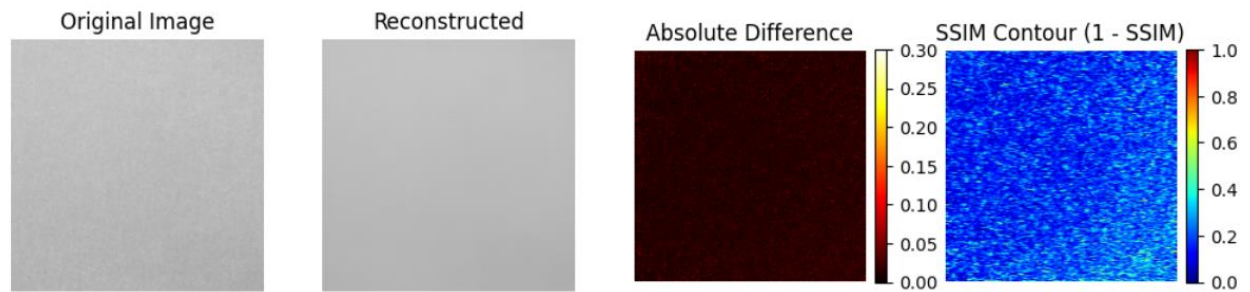
Đề tài: Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải



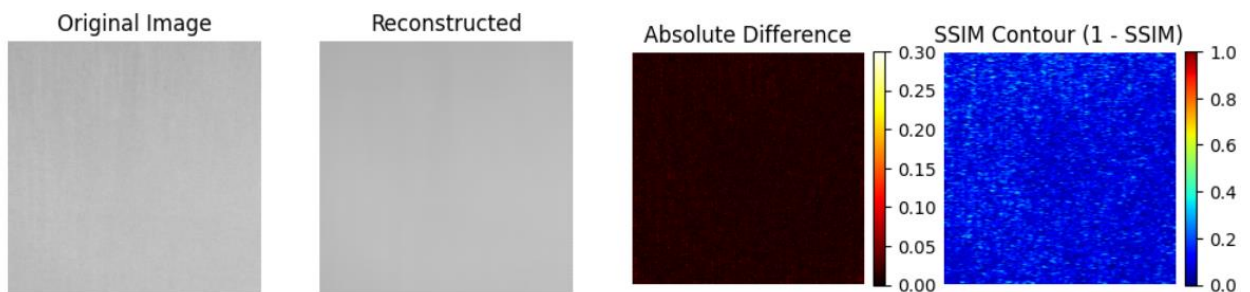
=== Reconstruction Quality Metrics ===
SSIM: 0.9242
Max Difference: 0.0836



=== Reconstruction Quality Metrics ===
SSIM: 0.9217
Max Difference: 0.0586

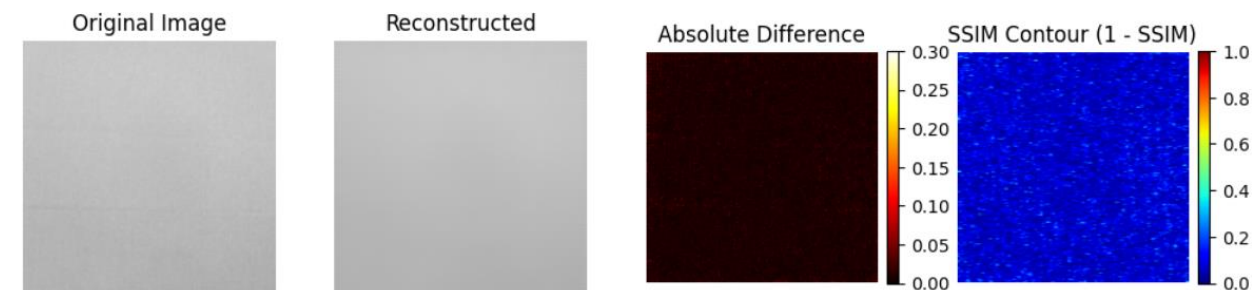


=== Reconstruction Quality Metrics ===
SSIM: 0.8196
Max Difference: 0.1176

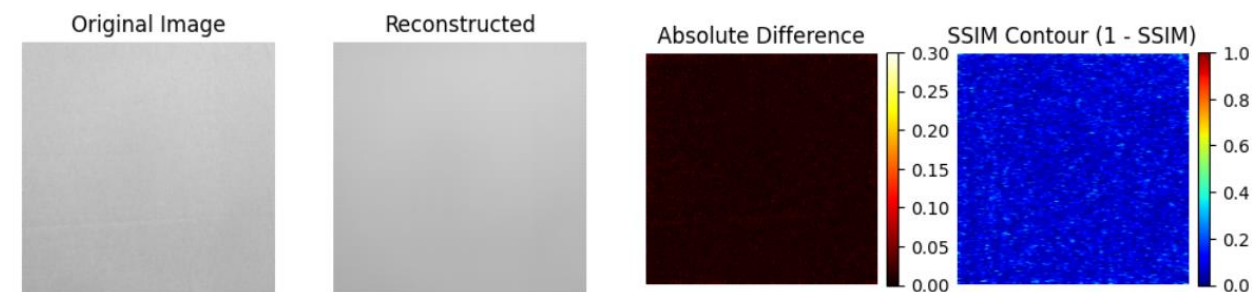


=== Reconstruction Quality Metrics ===
SSIM: 0.8948
Max Difference: 0.0664

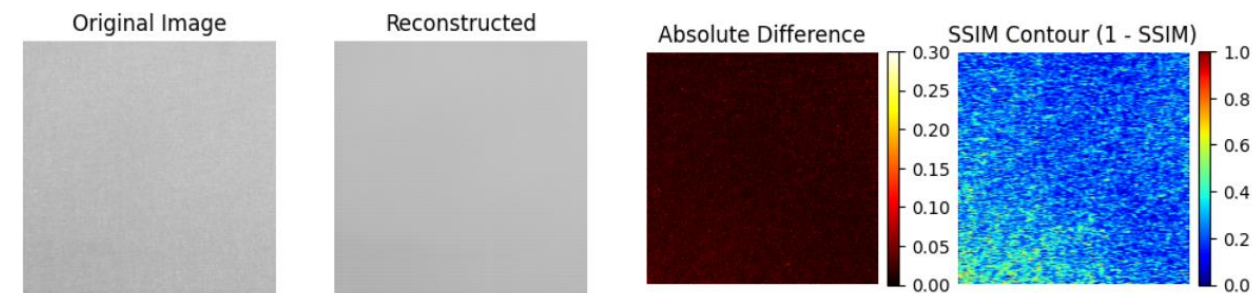
Đề tài: Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải



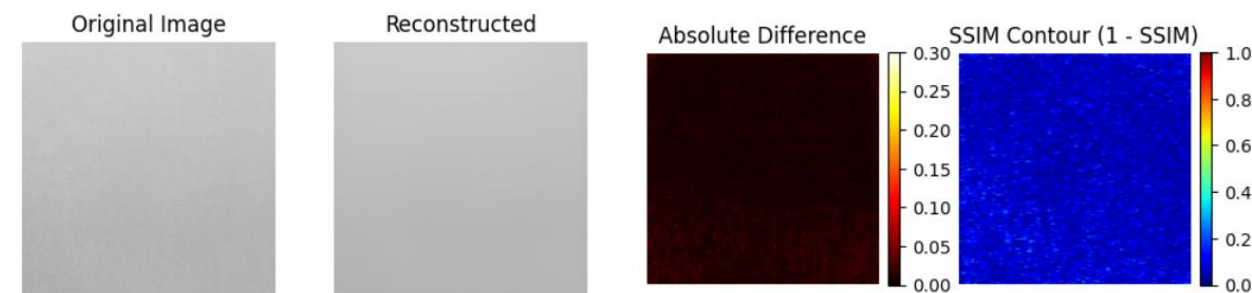
=== Reconstruction Quality Metrics ===
SSIM: 0.9278
Max Difference: 0.0724



=== Reconstruction Quality Metrics ===
SSIM: 0.9189
Max Difference: 0.0745

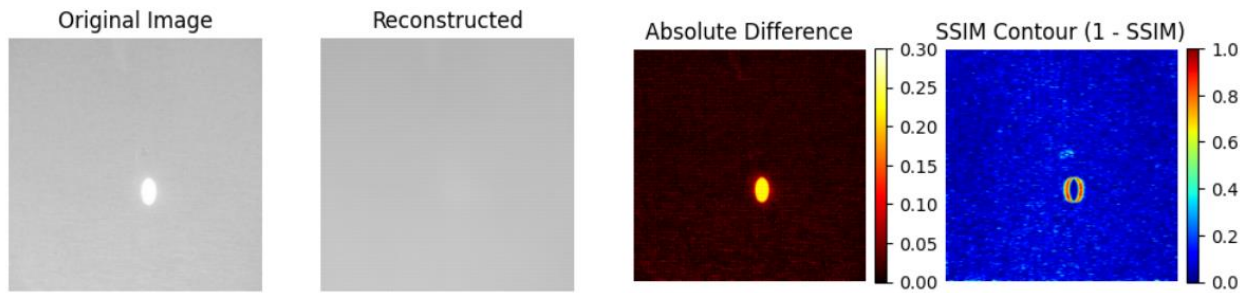


=== Reconstruction Quality Metrics ===
SSIM: 0.7591
Max Difference: 0.1198

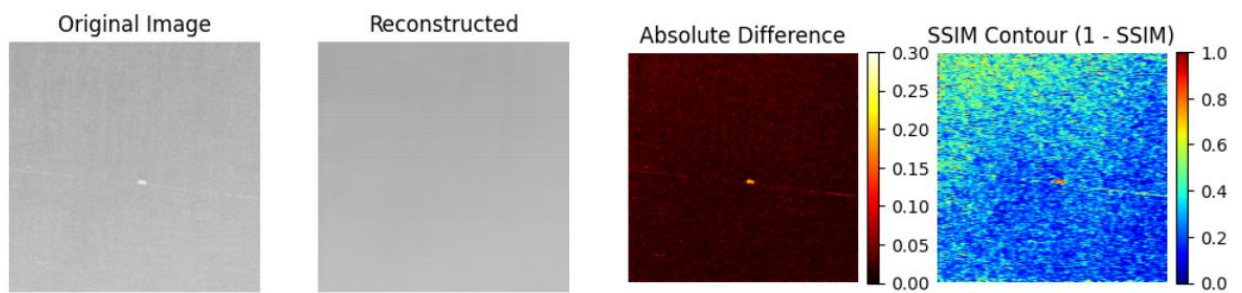


=== Reconstruction Quality Metrics ===
SSIM: 0.9403
Max Difference: 0.0584

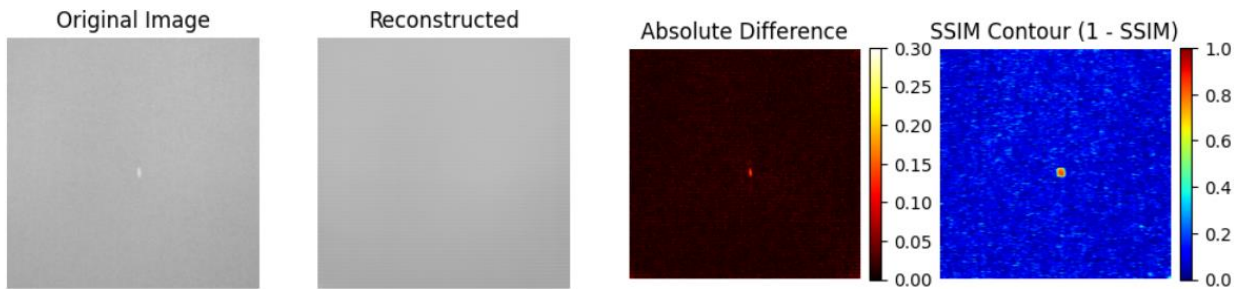
Phụ lục 2: Ảnh kết quả thu thập của nhóm có lỗi để chọn ngưỡng phân loại



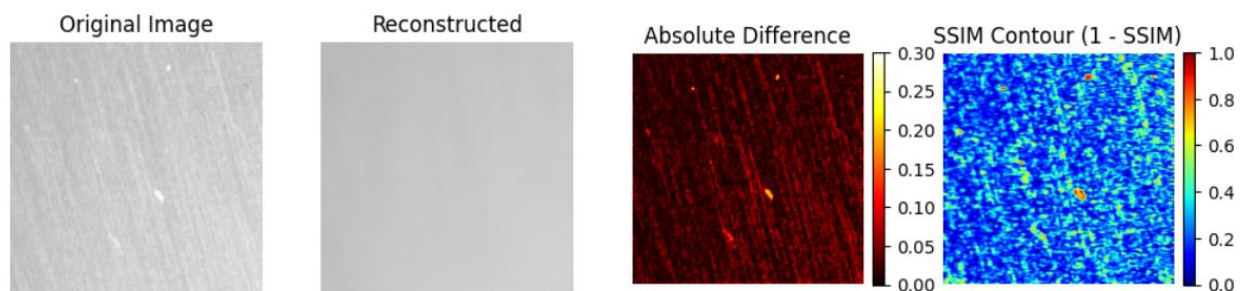
=== Reconstruction Quality Metrics ===
SSIM: 0.9276
Max Difference: 0.2365



=== Reconstruction Quality Metrics ===
SSIM: 0.7221
Max Difference: 0.2447

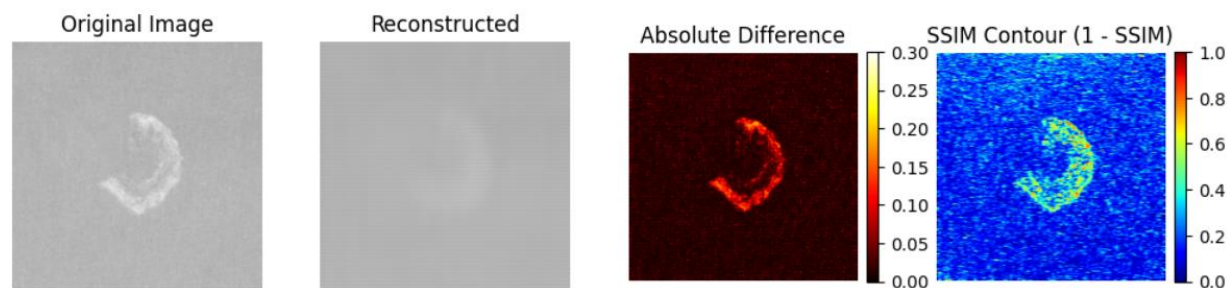


=== Reconstruction Quality Metrics ===
SSIM: 0.9098
Max Difference: 0.1587

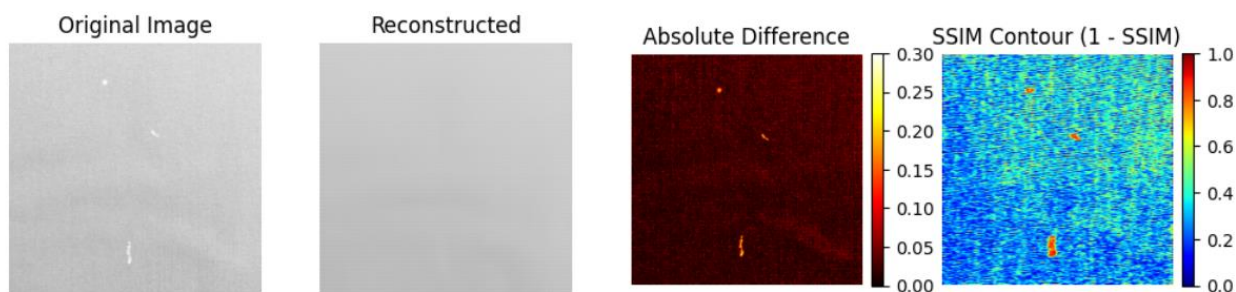


=== Reconstruction Quality Metrics ===
SSIM: 0.7567
Max Difference: 0.2201

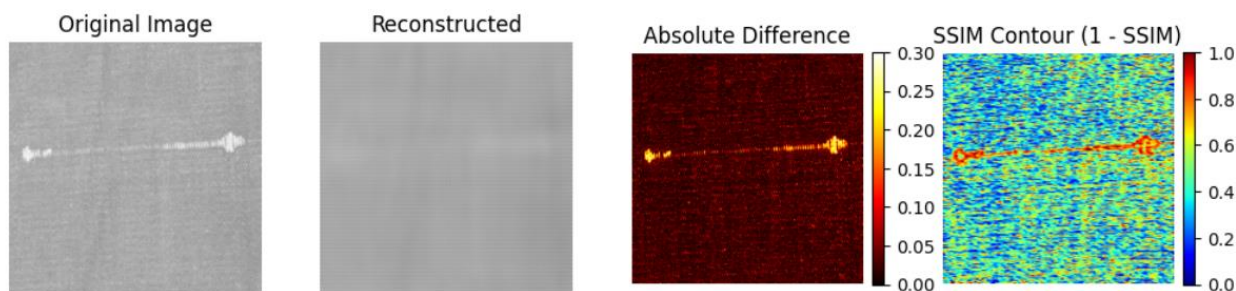
Đề tài: Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải



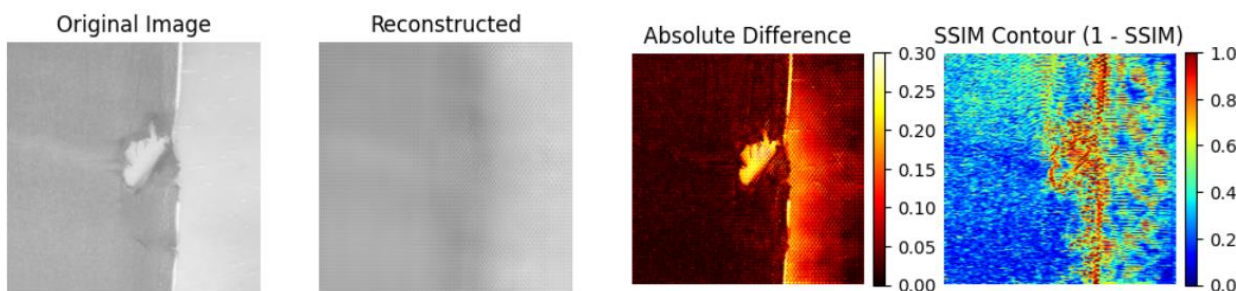
=== Reconstruction Quality Metrics ===
 SSIM: 0.8200
 Max Difference: 0.2083



=== Reconstruction Quality Metrics ===
 SSIM: 0.6803
 Max Difference: 0.2369

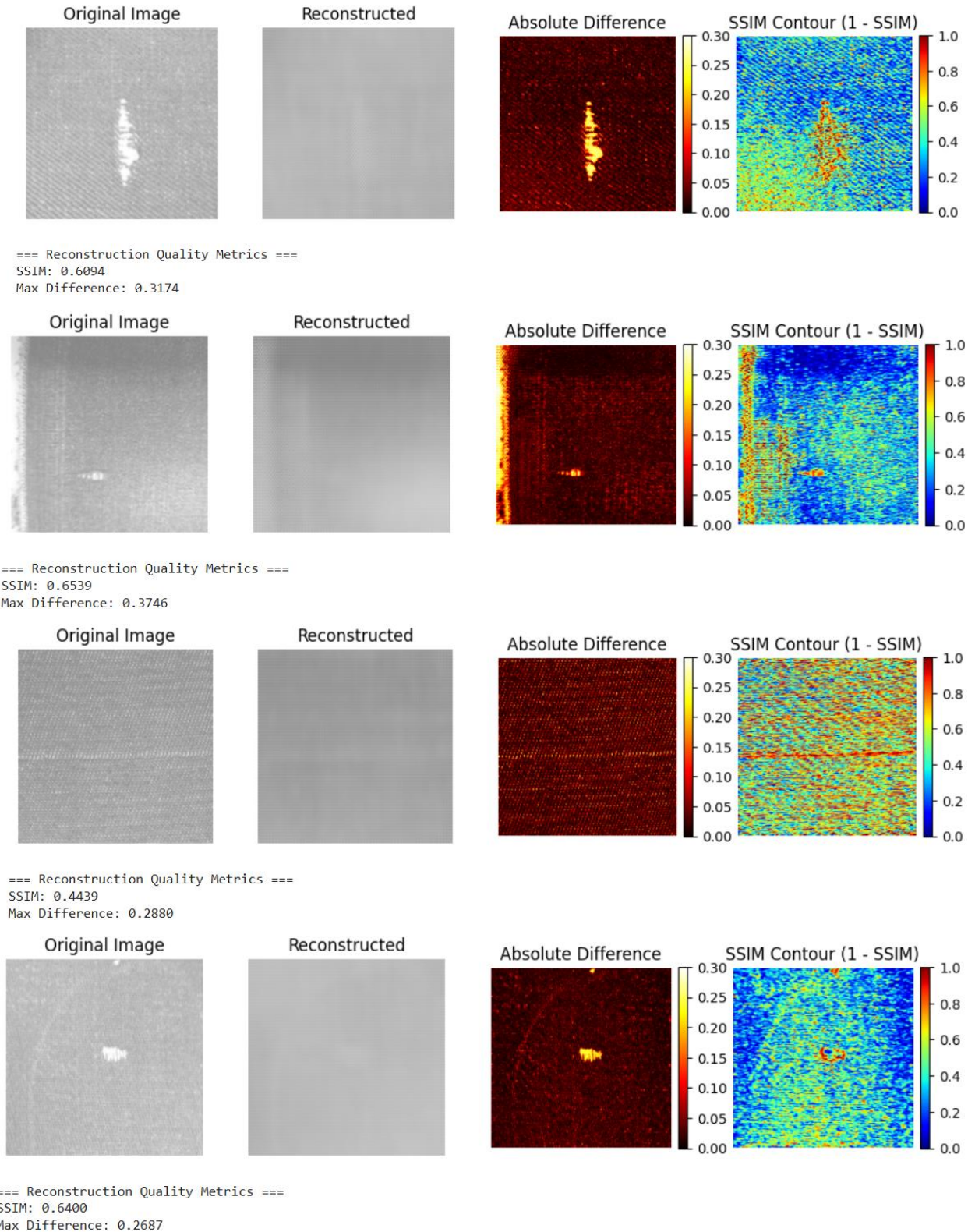


=== Reconstruction Quality Metrics ===
 SSIM: 0.5707
 Max Difference: 0.3077

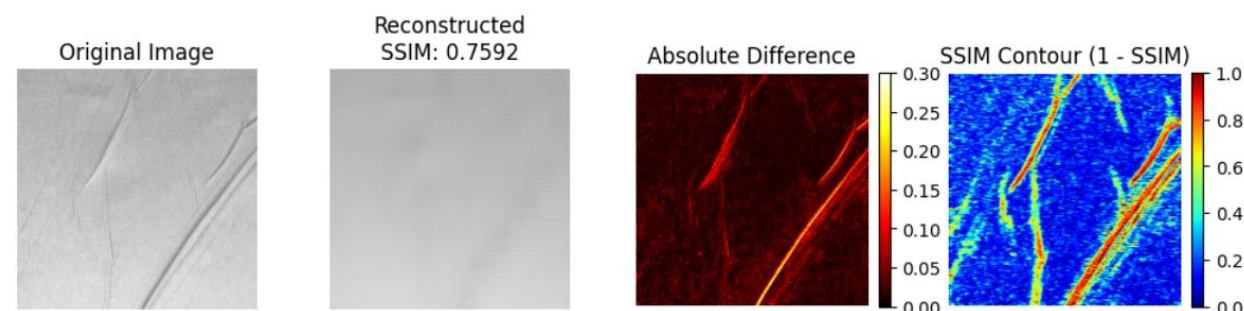
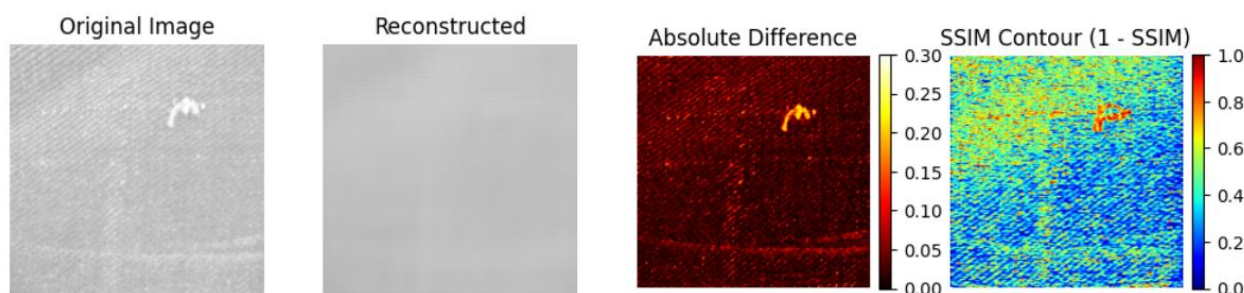
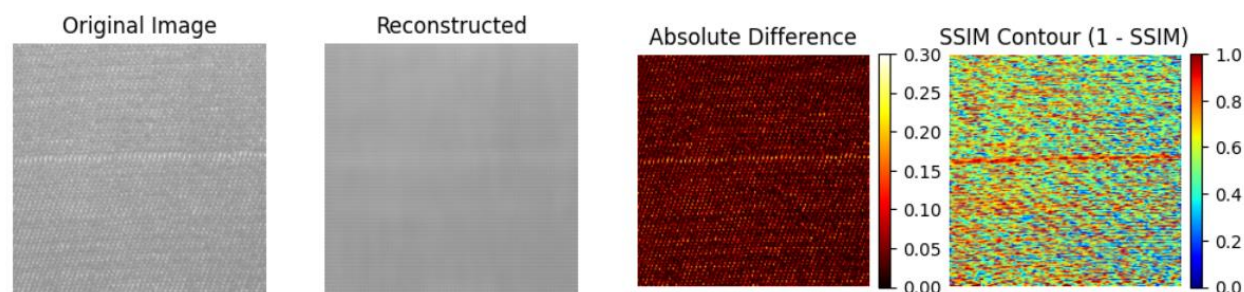
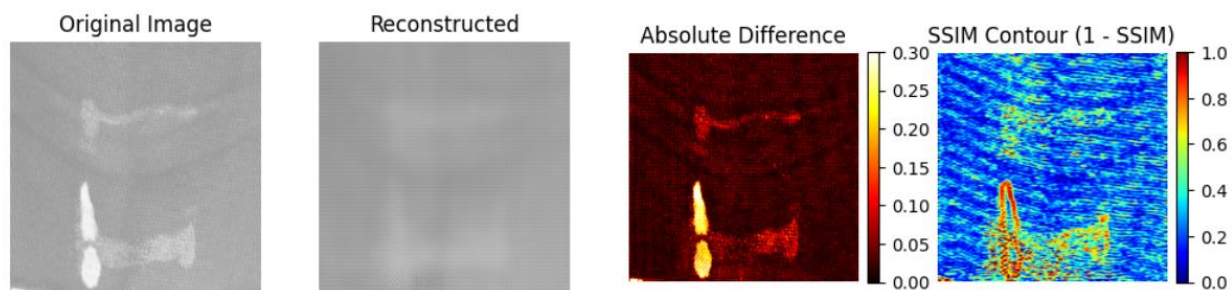


=== Reconstruction Quality Metrics ===
 SSIM: 0.6271
 Max Difference: 0.3464

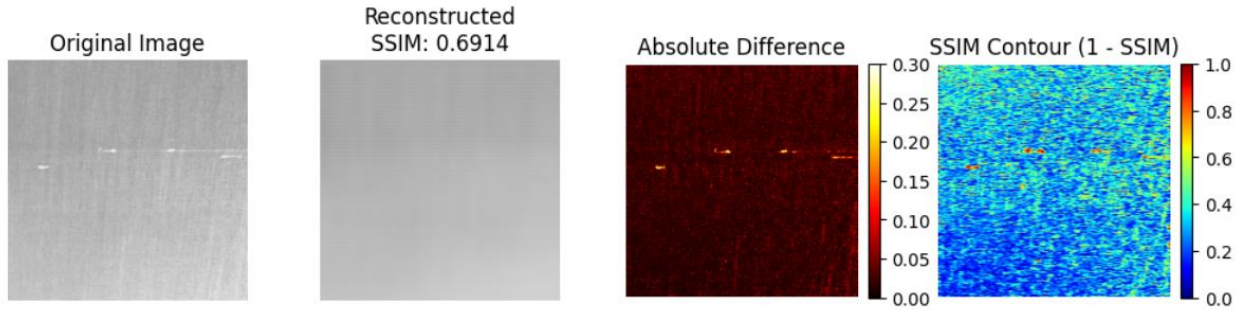
Đề tài: Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải



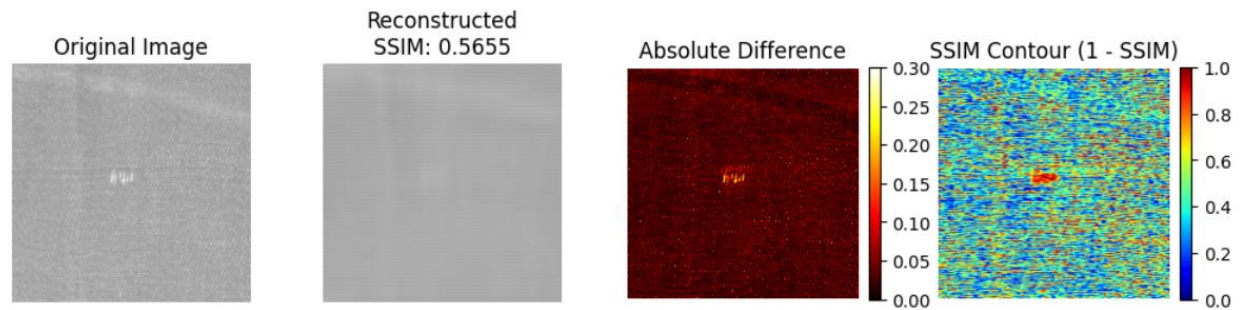
Đề tài: Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải



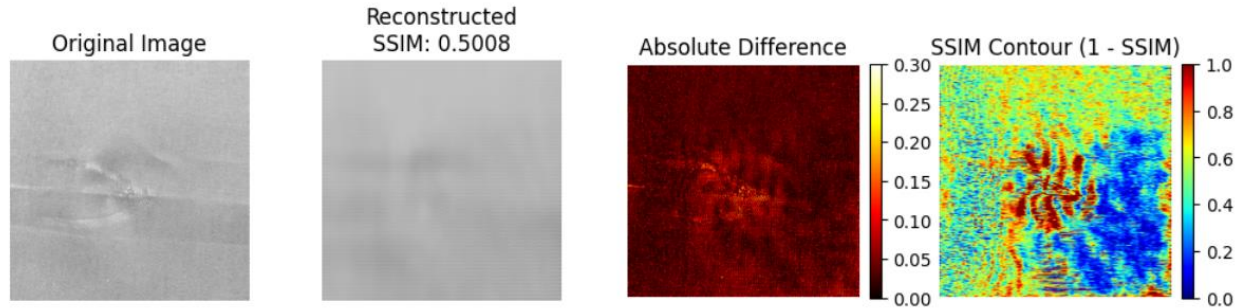
Đề tài: Nghiên cứu ứng dụng mạng nơ-ron tự mã hóa (Autoencoder) trong phát hiện khuyết tật sản phẩm vải



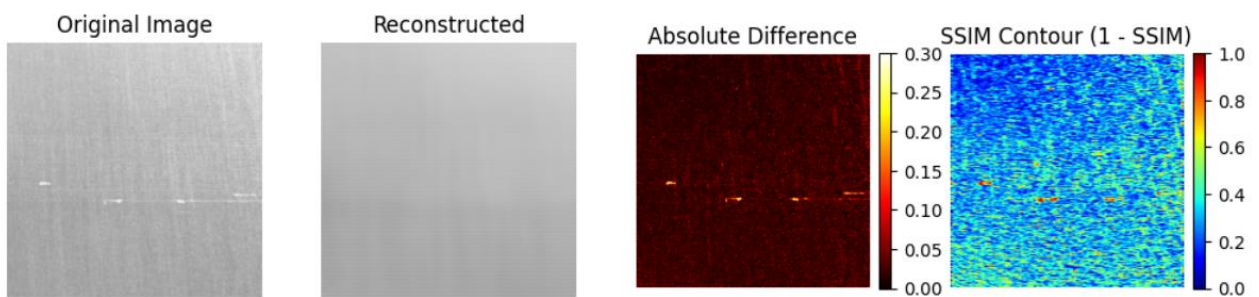
=== Reconstruction Quality Metrics ===
SSIM: 0.6914
Max Difference: 0.2905



=== Reconstruction Quality Metrics ===
SSIM: 0.5655
Max Difference: 0.2456



=== Reconstruction Quality Metrics ===
SSIM: 0.5008
Max Difference: 0.2753



=== Reconstruction Quality Metrics ===
SSIM: 0.6906
Max Difference: 0.2953