

엣지 컴퓨팅 환경에서 영상 처리의 성능 향상을 위한 얼굴 인식 기반 영상 검색 시스템

최진수*¹, 김장원*², 이석훈*³

Face Recognition-based Video Retrieval System for Improving Video Processing Performance in Edge Computing Environment

Jinsu Choi*¹, Jangwon Gim*², and Sukhoon Lee*³

이 성과는 2020년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임
(No. NRF-2020R1C1B6008624)

요 약

영상 데이터를 활용하는 인공지능 모델은 많은 양의 학습 데이터가 요구되며, 데이터 확보를 위해 CCTV와 같이 영상을 수집하고 검색 및 추출 가능한 시스템이 필요하다. 기존의 영상 검색 시스템은 주로 중앙 집중화된 구조의 영상 처리 방식으로 고성능의 컴퓨팅 리소스를 요구하며, 높은 처리 시간 및 전송 지연 등의 문제를 발생시킨다. 이러한 문제를 해결하기 위하여, 본 논문은 엣지 컴퓨팅 환경에서 얼굴 인식 기반 영상 검색 시스템을 제안한다. 이를 위하여, 제안 시스템은 영상 수집 시 엣지 디바이스에서 영상의 얼굴 탐지 후 얼굴 특징을 추출한다. 영상 검색은 사용자가 검색하고자 하는 사람의 얼굴 사진을 입력하면 HDBSCAN으로 분류된 얼굴들이 출현하는 영상 클립을 제공해 주는 방식으로 구현된다. 평가를 통해, 제안 시스템은 오류 없이 영상이 검색됨을 검증하였으며, 엣지 디바이스의 수가 많아질수록 영상 처리 성능이 향상됨을 확인하였다.

Abstract

An artificial intelligence model using video data requires a large amount of learning data, and a system that can collect, retrieval, and extract videos, such as CCTV, is required to secure data. Existing video retrieval systems are mainly centralized video processing methods, which require high-performance computing resources and cause problems such as high processing time and transmission delay. To solve this problem, this paper proposes a face recognition based video retrieval system in an edge computing environment. To this end, the proposed system extracts facial features after detecting the face of the video in the edge device when collecting the video. Video retrieval is implemented in such a way that when a user inputs a face photo of a person to be searched for, a video clip in which faces classified by HDBSCAN appear appears is provided. Through the evaluation, it was verified that the proposed system retrieved video without errors, and it was confirmed that the video processing performance improved as the number of edge devices increased.

Keywords

edge computing, face recognition, video retrieval system, video processing performance

* 군산대학교 소프트웨어학부 (*^{2,3} 교신저자) • Received: Feb. 09, 2023, Revised: Mar. 15, 2023, Accepted: Mar. 18, 2023
- ORCID¹: <https://orcid.org/0000-0002-5431-5273> • Corresponding Author: Sukhoon Lee and Jangwon Gim
- ORCID²: <https://orcid.org/0000-0002-4480-7944> School of Software, Kunsan National University, Korea
- ORCID³: <https://orcid.org/0000-0002-3390-5602> Tel.: +82-63-469-8914, Email: {leha82, jwgim}@kunsan.ac.kr

1. 서 론

CCTV와 같은 감시 카메라는 공공 안전, 시설 관리, 보안 및 감시 목적을 위해 보편화되어 대다수의 장소에 설치되며 지속적으로 영상 데이터를 수집한다[1]. 이렇게 확보된 영상 데이터는 영상 분석을 통해 객체 식별 및 추적, 보행자 감지, 의심 행동 감지, 차량 궤적 이상 식별, 교통 이상 예측, 영상 요약 등 다양한 분야에서 활용될 수 있다[2][3].

기존의 영상 분석은 수동적인 영상 분석으로 사람이 직접 영상을 재생하여 일련의 프레임에 등장한 객체와 해당 프레임의 상황을 인지한다. 이러한 방법은 확보된 영상을 모두 확인하는 과정에서 많은 시간과 인적비용이 필요하므로 자동화된 영상 분석을 위해 인공지능 모델이 요구된다[4].

인공지능 모델을 활용한 영상 기반의 시스템은 다양한 분야에서 연구되고 있으며 특히, 인물의 식별 정보인 얼굴의 특징을 활용한 얼굴 인식 연구가 활발히 진행되고 있다[5]-[7]. 하지만 인공지능 모델 구축을 위해서는 많은 양의 학습 데이터가 요구된다. 또한, 데이터 확보를 위해 지속적으로 영상을 수집하고 확보된 영상 속에서 학습에 필요한 영상을 검색하고 추출하기 위한 시스템이 필요하다.

이를 위하여, [8]은 기존의 확보된 영상 데이터베이스에서 특정 객체를 기반으로 영상을 검색하는 시스템을 제안한다. [9]는 객체 검색 시 쿼리 객체를 포함하는 영상 클립을 인덱싱하며, 이를 위해 영상을 프레임 집합으로 나누는 과정과 프레임 단위의 객체 검출 및 분석 과정으로 구분하여 객체가 포함된 영상을 추출한다.

한편, 영상 검색 시스템은 대부분 영상의 저장 및 인덱싱을 위하여 클라우드 컴퓨팅 환경을 기반으로 개발된다[10][11]. 특히, CCTV와 같은 실시간 영상 수집 시스템들은 각 카메라에서 촬영된 영상을 클라우드 서버로 전송하고 수집된 영상을 집중적으로 분석한다.

하지만, 이러한 클라우드 컴퓨팅 환경에 기반한 영상 수집 및 분석은 다음과 같이 두 가지 문제점이 존재한다. 먼저, 데이터 전송 및 수집에 있어서 영상 데이터는 고용량 하드웨어와 높은 네트워크

비용을 요구한다는 것이다. 또한, 해상도 및 초당 프레임 수, 픽셀 분포와 같은 구성 요소에 따라 영상 데이터 용량이 급격히 증가할 수 있으므로, 더 좋은 품질의 데이터를 분석하기 위해서는 더 높은 비용이 요구된다[12]. 두 번째로, 클라우드 서버에서 수집된 영상들의 분석 역시 고성능 컴퓨팅 사양과 병렬처리 등의 복잡한 알고리즘 등이 요구된다는 것이다. 또한, 데이터 규모에 따라 높은 처리 시간과 전송 지연 및 병목현상을 발생시킨다[13].

이러한 문제의 해결을 위해, 엣지 컴퓨팅을 활용한 연구들이 진행되었다[14][15]. 엣지 컴퓨팅은 기존의 클라우드 컴퓨팅이 가지는 중앙 집중화된 구조와 달리 작업 태스크 일부를 데이터 소스에 가까운 엣지 디바이스에서 제한된 리소스를 활용하여 분산 처리하는 구조를 지닌다. 이를 통해 엣지 디바이스는 영상을 수집함과 동시에 해당 영상을 분석한 결과를 클라우드 서버로 전송하므로, 클라우드 서버에서 모든 데이터를 분석해야 하는 부담을 줄일 수 있다.

따라서 본 논문은 엣지 컴퓨팅 환경에서 영상 처리의 성능 향상을 위한 얼굴 인식 기반 영상 검색 시스템을 제안한다. 제안 시스템은 수집된 영상을 기반으로 특정 대상의 얼굴을 인식하여 출현한 영상을 검색하고 해당 영상을 추출한다. 이때, 제안 시스템의 영상 처리 성능 향상을 위해 클라우드 서버에서 영상 내 출현한 얼굴을 인식하고 분석하는 대신, 각 엣지 디바이스의 리소스를 활용하여 각 디바이스가 수집한 영상을 직접 분석한다. 제안 시스템의 평가를 위하여 본 논문은 영상 검색 성능을 평가하고, 제안 시스템이 원활하게 동작함을 검증한다. 또한, 엣지 컴퓨팅 환경과 클라우드 컴퓨팅 환경의 비교 평가를 통하여 제안 시스템의 영상 처리 성능이 향상됨을 검증한다.

본 논문의 구성은 다음과 같다. 2장은 관련 연구를 소개하고, 3장에서는 제안 시스템의 구조와 얼굴 인식 및 영상 검색 기법을 기술한다. 4장에서는 구현 결과를 보이며, 5장에서는 실험 및 평가를 기술한다. 마지막으로 6장에서는 결론과 향후 연구를 기술한다.

II. 관련 연구

2.1 CCTV 기반 영상 검색 시스템

[16]은 지능형 CCTV 시스템에서 인물의 활동 검색을 위한 클라우드 컴퓨팅 기반의 영상 검색 시스템을 제안한다. 먼저, 수집된 영상은 실시간 스트리밍 형태로 클라우드 서버로 전송되어 합성곱 신경망과 잔차 신경망을 통해 영상에서 출현한 인물을 검출하고 활동을 분류한다. 또한, 온톨로지를 활용하여 해당 영상의 어노테이션을 생성한다. 이후, 특정 활동에 대한 질의와 어노테이션을 통한 인물 활동 기반의 영상 검색이 이루어진다.

[17]은 다중 CCTV로부터 접수된 실종자를 인식하고 추적하기 위한 인상착의 기반의 실시간 영상 검색 시스템을 제안한다. 먼저, 수집된 원본 영상은 원격지의 실시간 스트리밍 서버로 전송되어 관리되며 검색 요청에 따라 YOLOv3을 활용한 인물 탐지와 입력받은 인상착의를 통한 HSV 색상 값 분석으로 특정 인물이 출현한 영상을 검색 및 추출한다.

[18]은 CCTV로부터 수집된 영상에서 객체 기반의 영상 검색을 위한 시스템 아키텍처를 제안한다. 영상 검색을 위하여 확보된 영상에서 YOLO를 통해 일련의 프레임에서 등장하는 객체를 감지하고 등장 횟수를 포함한 메타데이터를 생성한다. 이후, Apache Solr를 통해 영상 데이터베이스에서 질의어에 해당하는 영상을 검색한다.

이러한 연구들은 인물의 인상착의, 활동 및 객체 수 등을 특징으로 영상을 검색한다. 하지만, 해당 특징들은 영상에 출현한 특정 객체를 식별하는 특성이 아니므로 얼굴과 같이 사람을 식별하는 특징으로 활용할 수 없다.

따라서, 본 논문은 CCTV와 같은 영상 시스템에서 사람의 얼굴 인식 기법을 통하여 특정 인물이 출현한 영상을 검색할 수 있는 시스템 개발을 목표로 한다.

2.2 엣지 컴퓨팅 기반 영상 처리 기법

[19]는 엣지 디바이스와 클라우드 서버가 협업하

는 모바일 엣지 컴퓨팅 기반의 얼굴 인식을 위한 연구이다. 클라우드 컴퓨팅 환경에서는 엣지 디바이스가 확보한 영상을 클라우드 서버로 전송하므로 네트워크 대역폭에 높은 오버헤드를 발생시킨다. 해당 연구에서는 얼굴 검출 및 인식 과정을 엣지 디바이스가 수행하여 클라우드 서버로 보내는 대규모 데이터 전송을 최소화한다.

[20]은 엣지 컴퓨팅 기반의 영상 스트리밍 전송 최적화 기법을 제안한다. 엣지 디바이스가 확보된 영상을 원격지의 서버로 전송할 때, 고품질의 영상을 전송하기 위해 높은 대역폭이 요구된다. 이는 대역폭의 상태에 따라 영상의 품질이 변화되는 문제를 야기하므로, 선명도 비교, 물체 감지, 객체 위치 예측 등의 알고리즘들을 통하여 저대역폭에서도 고해상도 영상을 전송할 수 있는 기법을 제안한다.

[21]은 지능형 교통 시스템에서 발생하는 실시간 영상 데이터를 처리할 수 있는 엣지 컴퓨팅 기반의 영상 전처리 기법을 제안한다. 해당 연구는 엣지 디바이스 및 클라우드 서버 간 영상 전송 데이터 규모를 줄이기 위하여 영상 세분화와 시공간적 관심 지점을 통해 영상 속 키 프레임을 추출하고 중복되거나 불필요한 프레임을 제거한다. 이를 통해, 네트워크 대역폭을 줄이고 클라우드 서버의 컴퓨팅 리소스와 스토리지 용량을 확보한다.

이러한 연구들은 중앙 집중화된 구조가 지니는 높은 데이터 전송 비용과 영상 처리 지연 문제를 해결하기 위하여 엣지 컴퓨팅 환경에서 영상 처리 성능 향상을 목표로 연구되었다.

본 논문은 지속적인 영상 처리와 영상 검색을 위해 엣지 컴퓨팅 환경에서 시스템을 구성하고 영상 처리 성능 향상을 목표로 한다.

III. 제안 시스템

이 장에서는 제안 시스템의 구조를 기술한다. 먼저, 제안 시스템의 영상 처리 모듈의 구조를 기술하고, 얼굴 클러스터링 과정을 설명한다. 이후, 영상 검색을 위한 데이터베이스 구조를 정의하고 영상 검색 웹 서비스의 구조를 기술한다.

3.1 제안 시스템 구조

그림 1은 제안 시스템의 구조를 보인다. 제안 시스템의 구조는 크게 엣지(Edge), 서버(Server), 애플리케이션(Application) 계층으로 구성되어 있으며, 각 계층의 구성 요소와 주요 기능은 다음과 같다.

엣지 계층은 카메라와 영상 처리 모듈(Video processing module)이 포함된다. 배포된 각 엣지 디바이스는 영상 처리 모듈을 통해 실시간 영상 데이터를 수집하고, 분석하여 영상에 등장하는 얼굴의 특징을 추출하고 데이터베이스로 전송한다.

서버 계층은 얼굴 분석 모듈(Face analysis module)과 데이터베이스(Database)로 구성된다. 먼저, 얼굴 분석 모듈은 수집된 얼굴 특징 데이터를 기반으로 클러스터링하여 얼굴 클러스터를 최신화한다. 데이터베이스는 영상 스토리지(Video storage)와 특징 스토리지(Feature storage)로 구성되며, 각각 수집된 영상 데이터와 얼굴 특징 데이터를 관리한다.

애플리케이션 계층은 영상 검색 웹 서비스(Video retrieval web service)를 제공한다. 사용자 검색 요청에 따라 확보된 영상 데이터베이스에서 특정 인물의 출현 영상과 메타데이터를 추출하여 검색된 영상을 사용자에게 제공한다.

3.2 영상 처리 모듈

엣지 계층에서 엣지 디바이스는 영상 처리 모듈을 통해 영상을 수집하고 확보된 영상을 분석하여 서버 계층의 데이터베이스로 전송한다. 영상 처리 모듈은 목적에 따라 크게 영상 스트림(Video stream), 영상 인코딩(Video encoding), 얼굴 특징 추출(Face feature extraction)로 구분된다.

영상 스트림은 엣지 디바이스가 카메라 센서를 통해 영상 데이터를 수집하는 과정이다. 지속적인 영상 수집 및 분석을 위하여 수집되는 영상의 해상도는 960*720, 초당 프레임 수는 24로 구성한다.

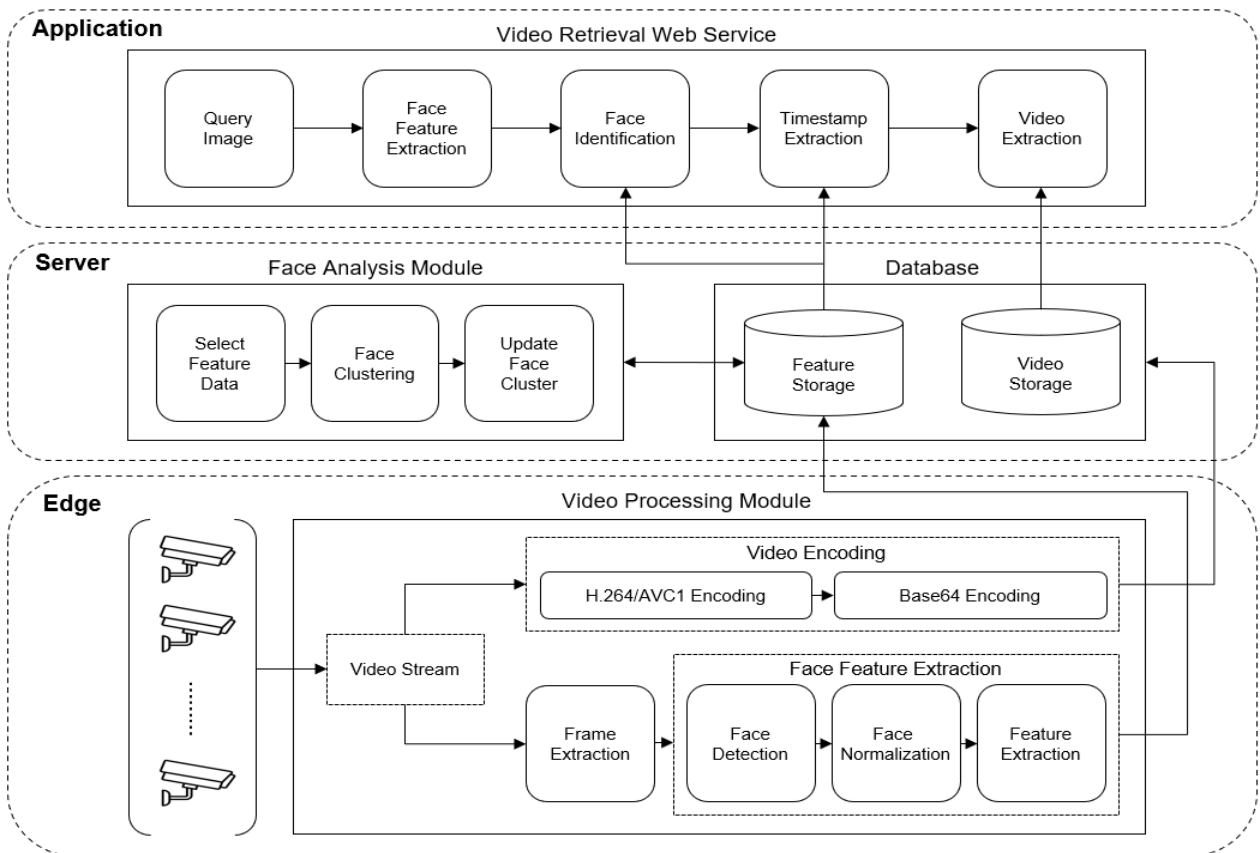


그림 1. 제안 시스템의 구조
Fig. 1. An architecture of proposed system

영상 인코딩은 수집된 영상을 압축하고 서버 계층의 영상 스토리지로 전송하는 과정이다. 먼저, 서버 계층으로 전송하는 영상 데이터의 규모를 줄이기 위하여 H.264/AVC1 인코딩을 통해 영상을 손실 압축한다. 손실 압축된 영상은 Base64 인코딩을 통해 문자열 데이터로 변환되어 서버 계층의 영상 스토리지로 전송된다.

얼굴 특징 추출은 수집된 영상에서 출현하는 얼굴을 찾고, 얼굴 특징 데이터를 추출하여 서버 계층의 특징 스토리지로 전송하는 과정이다. 그림 2는 얼굴 특징 추출 과정을 보인다.

프레임 추출(Frame extraction)은 영상을 일련의 프레임으로 추출하는 과정이다. 짧은 간격의 프레임은 움직임에 의한 블러링이 발생할 수 있으므로 주어진 영상 클립에서 1초 간격으로 프레임을 추출한다.

얼굴 검출(Face detection)은 프레임에 등장하는 각 얼굴 영역을 검출하는 과정이다. 얼굴 검출을 위하여 OpenCV의 Haar Cascade 알고리즘을 활용한다. 해당 알고리즘은 다른 얼굴 검출 알고리즘보다 속도가 빠르며 요구되는 컴퓨팅 리소스가 적으므로 제안 시스템의 얼굴 검출 알고리즘으로 활용한다[22].

얼굴 정규화(Face normalization)는 검출된 얼굴이 정면에 위치하도록 정렬하고 얼굴 영역의 크기를 조정하는 과정이다. 얼굴 정렬은 DeepFace의 Haar Cascade 기반 얼굴 정렬 알고리즘을 활용한다. 해당 알고리즘은 눈 검출기를 통해 양쪽 눈을 찾고 수평이 되도록 얼굴을 회전한다. 얼굴 정렬 이후, 얼굴 영역의 크기를 80x80으로 조정한다.

특징 추출(Feature extraction)은 앞서 정규화된 얼굴 영역으로부터 128차원의 얼굴 특징 데이터를 추출하는 과정이다. 제안 시스템은 얼굴 영역의 특징 추출을 위해 dlib의 Facial Landmark 모델을 활용한

다. 해당 모델은 미리 학습된 심층 신경망으로 얼굴 윤곽, 눈썹, 눈, 코, 입 등의 얼굴 구성 요소를 통해 128차원의 얼굴 임베딩 벡터를 추출한다. 각 얼굴 영역에서 추출된 얼굴 임베딩 벡터는 바이너리 형태로 서버 계층의 특징 스토리지에 전송된다.

3.3 얼굴 분석 모듈

서버 계층의 얼굴 분석 모듈은 생성된 얼굴 특징 데이터를 기반으로 얼굴 클러스터를 최신화한다. 얼굴 분석 모듈은 얼굴 클러스터의 수를 알 수 없는 가변적인 환경으로 밀도 기반의 클러스터링 기법인 HDBSCAN(Hierarchical Density-Based Spatial Clustering of Applications with Noise)을 활용한다. HDBSCAN은 기존의 DBSCAN 알고리즘에서 계층적 클러스터링 방법을 적용한 기법으로, 비교적 시간 복잡도가 높지만 더욱 복잡한 형상을 가지는 데이터에서 다양한 밀도의 클러스터를 식별할 수 있다는 장점이 있다[23].

그림 3은 128차원의 얼굴 특징 데이터를 PCA(Principal Component Analysis)를 통해 2차원으로 축소된 클러스터링 결과를 보인다.

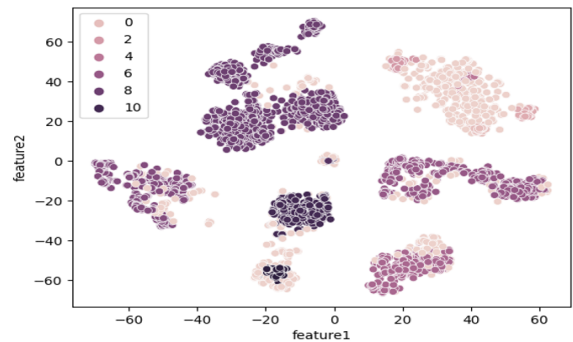


그림 3. 얼굴 클러스터링 결과의 예
Fig. 3. An example of face clustering

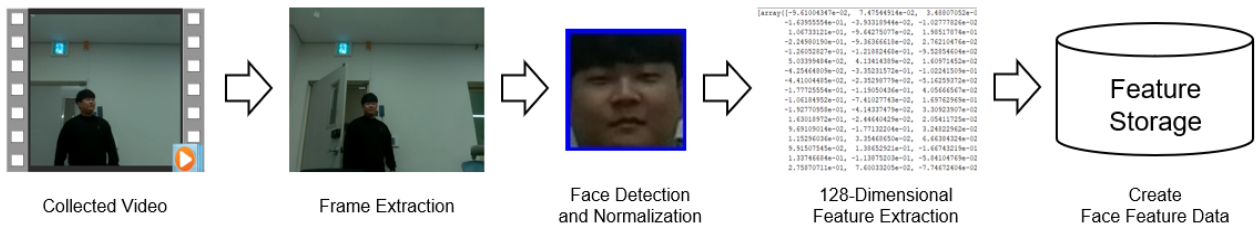


그림 2. 얼굴 특징 추출 프로세스
Fig. 2. Extraction process of face feature

3.4 데이터베이스 구조

데이터베이스는 엣지 계층의 영상 처리 모듈을 통해 생성된 데이터를 관리하며 영상 스토리지와 특징 스토리지로 구분된다. 먼저, 영상 스토리지는 시계열 데이터베이스인 InfluxDB를 활용한다. InfluxDB는 NoSQL로 비정형 데이터인 영상 데이터 관리에 효율적인 특성이 있으므로, 기본 키인 타임스탬프를 기반으로 데이터 생성 및 조회에 집중할 수 있다.

특징 스토리지는 관계형 데이터베이스인 MySQL를 활용하여 수집된 영상에서 추출된 얼굴 특징 데이터와 해당 영상에 대한 정보를 관리한다. 표 1과 2는 각각 영상 스토리지와 특징 스토리지의 데이터 구조를 보인다.

표 1. 영상 스토리지의 데이터 구조
Table 1. Data structure for video storage

Key	Data type	Description
Timestamp	Datetime	Primary key indicating the video recording start time
Device	String	Device identification value
Video	String	Encoded 10 seconds video clip

표 2. 특징 스토리지의 데이터 구조
Table 2. Data structure for feature storage

Key	Data type	Description
Device	string	Device identification value
Timestamp	string	Timestamp data referencing the video
Label	integer	Face cluster class data
Frame	integer	Frame number in which the face appeared
Landmark	binary	128-dimensional face landmark feature data

3.5 영상 검색 웹 서비스

애플리케이션 계층의 영상 검색 웹 서비스는 웹 서비스를 통한 사용자의 검색 요청에 따라 질의 대상이 출현한 영상과 메타데이터를 추출하고, 검색 결과를 사용자에게 제공한다.

그림 4는 영상 검색 과정을 보인다. 먼저, 사용자는 영상 검색을 위해 특정 대상의 얼굴이 포함된 질의 이미지(Query image)를 제출한다. 이후, 질의 이미지를 통해 앞서 언급한 영상 처리 모듈의 얼굴 특징 추출 과정에 따라 질의 이미지에서 등장하는 얼굴의 특징 데이터를 추출한다.

얼굴 식별(Face identification)은 질의 대상의 얼굴 클러스터 부류 값을 찾는 과정이다. 질의 대상의 얼굴 특징 데이터를 추출하고 특징 스토리지에 저장된 데이터와 비교하여 가장 유사한 얼굴이 가지는 얼굴 클러스터 부류 값을 식별한다.

타임스탬프 추출(Timestamp extraction)은 질의 대상이 출현한 영상 클립을 참조하는 타임스탬프를 추출하는 과정으로, 영상 검색 시 입력받은 검색 기간을 포함하여 얼굴 식별 결과 값을 통해 특징 스토리지를 조회하여 타임스탬프를 추출한다.

영상 인덱싱(Video indexing)은 추출된 타임스탬프를 통해 질의 대상이 출현한 영상을 영상 스토리지에서 조회한다.

영상 추출(Video extraction)은 질의 대상이 출현한 영상 클립과 각 영상 클립에 대한 정보를 추출하는 과정이다. 추출된 영상 클립은 문자열로 구성되며 영상 디코딩을 통해 원본 영상으로 복원한다.

이후, 영상 검색 결과로 질의 대상이 출현한 원본 영상과 각 영상에 대한 디바이스 ID, 타임스탬프, 출현 시간, 출현 및 퇴장 시기를 검색 결과로 제공한다.

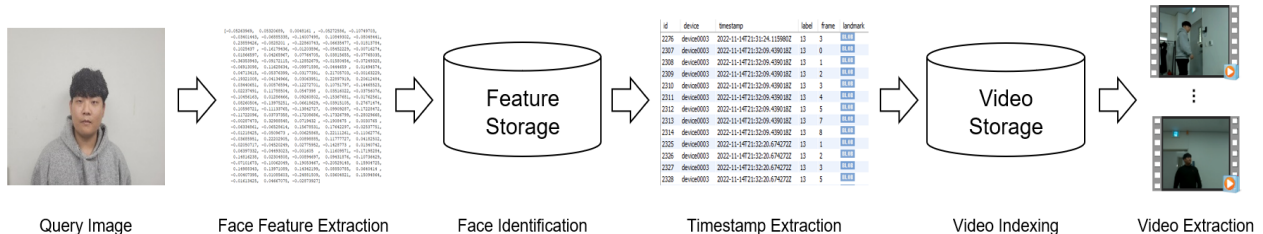


그림 4. 영상 검색 과정
Fig. 4. Video retrieval process

IV. 구현 결과

이 장은 엣지 컴퓨팅 환경에서 얼굴 인식 기반의 영상 검색 시스템을 실제로 구현하고 그 결과를 기술한다. 먼저, 표 3, 4, 5는 각각 엣지 디바이스, 데이터 스토리지, 서버의 구현 환경을 보인다.

표 3. 엣지 디바이스 구현 환경

Table 3. Edge device implementation environment

Features	Specification
Model	Raspberry Pi 3 B, Raspberry Pi 3 B+
OS	Raspbian(buster)
Sensor	Camera
Software	Python3, OpenCV, FFmpeg

표 4. 데이터 스토리지 구현 환경

Table 4. Data storage implementation environment

Features	Specification
OS	Microsoft Windows Server
CPU	Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz
SSD	500GB
Database	InfluxDB, MySQL

표 5. 서버 구현 환경

Table 5. Server implementation environment

Features	Specification
OS	Window 10 pro
CPU	Intel(R) Core(TM) i7-6700 CPU@ 3.40GHz
RAM	24GB
Software	Python3, Flask, Node.js, FFmpeg

구현된 제안 시스템의 실험을 위하여 사무실 내에서 엣지 디바이스 5대를 배포하여 영상을 수집하였다. 각 엣지 디바이스는 각기 다른 위치에 배포되어 제안 시스템의 영상 처리 모듈을 통해 영상 데이터와 얼굴 특징 데이터를 수집하였다. 그 결과, 10초 길이의 클립으로된 1,112개의 영상 데이터가 인코딩된 형태로 영상 스토리지(Influx DB)에 저장되었다.

그림 5는 영상 검색 웹 페이지의 구현 결과를 보인다. 사용자는 검색 요청을 위해 질의 대상의 얼굴이 포함된 정지 영상 파일을 제출하고 검색 기간을

설정한다. 이후, 제안 시스템의 영상 검색 과정에 따라 질의 대상이 출현한 영상과 각 영상에 대한 메타데이터 추출이 완료되면 영상 검색 결과 화면으로 이동한다.

그림 6은 영상 검색 결과 화면으로 질의 대상이 출현한 영상을 확인할 수 있다. 해당 화면에서 질의 대상이 출현한 영상을 재생할 수 있으며, 영상과 메타데이터는 압축 파일 형태로 내려받을 수 있다.



그림 5. 영상 검색 웹 페이지

Fig. 5. Video retrieval web page

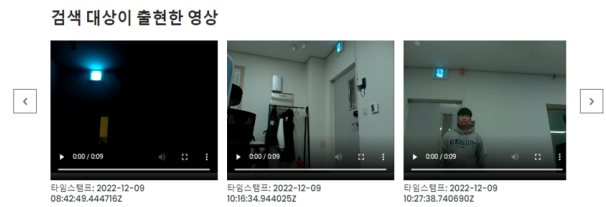
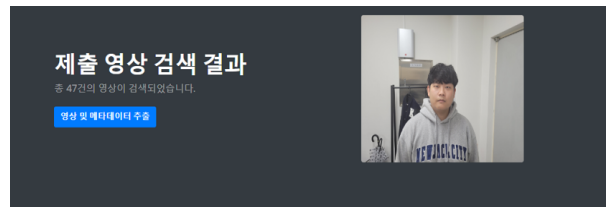


그림 6. 영상 검색 결과 화면

Fig. 6. Screenshot of video retrieval result

V. 실험 및 평가

이 장에서는 제안 시스템의 영상 검색 성능과 영상 처리 성능 향상을 검증하기 위하여 실험 및 평가를 수행한다.

5.1 영상 검색 성능 평가

이 절은 제안 시스템의 영상 검색이 원활하게 동작함을 검증하기 위하여 실제 영상 검색에 따른 성

능을 평가한다. 실험을 위해 앞서 수집된 1,112개의 영상 클립을 실험 데이터셋으로 구축한다.

실험 방법으로 먼저, 각 영상 클립에 존재하는 얼굴 특징 데이터를 추출하고, 각 특징 값에 대한 얼굴 클러스터를 생성한다. 실험 데이터셋에서 추출된 얼굴 특징 값은 총 4,295개이며 생성된 얼굴 클러스터는 총 13개이다. 실험을 위하여 출현한 5명의 인물로부터 얼굴이 포함된 정지 영상을 획득하여 제안 시스템의 영상 검색 웹 서비스를 통해 출현 영상을 검색한다. 이후, 영상 검색 결과에서 검색 오류를 확인하고 평균 영상 검색 시간과 10초 길이의 영상 클립에 대한 검색 시간을 구한다.

표 6은 5명의 인물에 대한 영상 검색 성능 결과를 보인다. 각 인물별로 5장의 얼굴 사진을 이용하여 영상을 검색하였다. 실험 결과 제안 시스템의 검색 오류는 없는 것으로 검증되었다. 한편, 영상 검색 시간은 추출하는 영상의 개수에 비례하며 하나의 영상 클립을 추출하기 위해 평균 2.19초의 검색 시간이 요구된다.

표 6. 영상 검색 성능 결과
Table 6. Results of video retrieval performance

Class	Number of input image	Average number of output clip	Average retrieval time (second)	Video clip per retrieval time (second)
1	5	97	216.69	2.23
2	5	120	294.08	2.45
3	5	81	215.52	2.66
4	5	47	89.60	1.90
5	5	121	208.24	1.72

5.2 영상 처리 시간 비교 평가

이 절은 제안 시스템의 영상 처리 성능을 평가하기 위해 엣지 및 클라우드 컴퓨팅 환경에서의 영상 처리 시간을 비교한다. 실험을 위해 10초 단위의 영상 클립 1,112개를 실험 데이터셋으로 구성한다.

실험 방법으로는 클라우드 컴퓨팅과 엣지 컴퓨팅 환경에서 영상이 주어질 때 제안 시스템의 얼굴 특징 추출 프로세스에 요구되는 시간을 각각 측정한다.

클라우드 컴퓨팅 환경에서는 1대의 클라우드 서버가 영상을 전송받아 처리할 때의 소요 시간을 측정하였으며, 엣지 컴퓨팅 환경에서는 엣지 디바이스가 병렬적으로 영상 수집 및 얼굴 특징 추출 프로세스를 수행하는 시간을 측정한다. 1대의 클라우드 서버와 3, 4, 5대의 엣지 디바이스가 각각 주어진 11,120초의 실험 데이터를 처리하였을 때 소요되는 누적 시간을 측정한다.

그림 7은 실험 결과로 엣지 및 클라우드 컴퓨팅 환경에서 영상 길이에 따른 영상 처리 시간을 나타낸다. 11,120초 길이의 전체 영상을 처리할 때 엣지 디바이스가 3대(edge-3), 4대(edge-4)일 경우 각각 5,517초와 4,257초의 처리 시간이 소요된다. 클라우드 서버(Cloud)에서는 3,506초의 처리 시간을 보이며 엣지 디바이스가 5대(edge-5)일 경우는 3,083초의 처리 시간이 소요된다. 해당 실험에서 edge-3과 edge-4일 때 cloud보다 소요되는 영상 처리 시간이 길었으며 edge-5일 경우에는 cloud보다 소요되는 영상 처리 시간이 더 짧았다. 이는 엣지 컴퓨팅 환경이 클라우드 컴퓨팅 환경보다 영상 처리 성능이 향상되는 분기점으로 제안 시스템 환경에서는 엣지 디바이스가 5대 이상 되어야 클라우드 서버보다 영상 처리의 성능이 향상됨을 보인다.

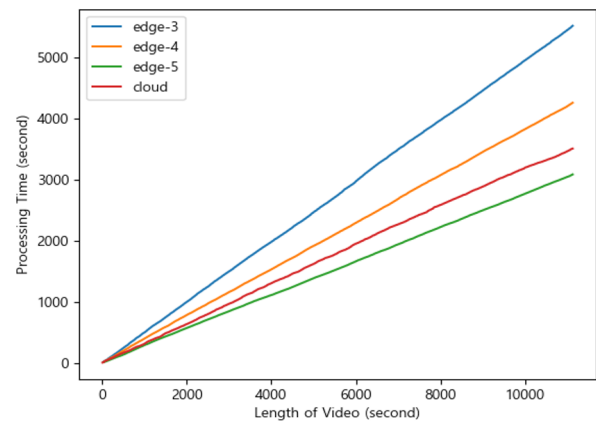


그림 7. 영상 처리 누적 시간 평가 결과
Fig. 7. An evaluation result of cumulative video processing time

표 7은 영상 처리 시간 측정 시 각 엣지 디바이스와 클라우드 서버가 하나의 영상 클립을 처리할 때의 평균 영상 처리 시간을 보인다.

표 7. 영상 클립 평균 처리 시간

Table 7. Average processing time for video clip

Device	Average video processing time(second)
Device 1	19.65
Device 2	8.96
Device 3	14.68
Device 4	15.78
Device 5	10.05
Cloud server	3.15

실험 결과 클라우드 서버의 경우 평균 3.15초의 영상 처리 시간을 보인다. 한편, 엡지 디바이스의 경우 각 엡지 디바이스들의 스펙 및 영상 처리 과정이 동일함에도 device 2가 평균 8.96초의 가장 낮은 영상 처리 시간을 보이며, device 1은 평균 19.65초의 가장 높은 영상 처리 시간을 보였다. 이는 영상 내 출현하는 사람의 수와 수집된 영상을 구성하는 픽셀 값이 밝기에 따라 영향을 받기 때문으로 분석된다.

VI. 결 론

기존의 클라우드 컴퓨팅 환경에서의 영상 검색 시스템은 다수의 카메라 센서로부터 수집된 영상을 원격지 서버로 전송하여 영상 분석이 이루어진다. 이는 중앙 집중화된 영상 처리로 인해 높은 처리 시간과 전송 지연을 발생시킨다.

따라서 본 논문은 엡지 컴퓨팅 환경에서 영상 처리 성능 향상을 위한 얼굴 인식 기반 영상 검색 시스템을 제안하였다. 제안 시스템은 클라우드 서버에서 영상을 전송 받아 분석하는 대신 엡지 디바이스가 영상 수집과 동시에 얼굴 특징을 직접 분석함으로써 병렬적인 영상 처리를 수행한다. 비교 평가를 통해 제안 시스템이 오류 없이 동작함을 검증하였으며 엡지 디바이스가 많아질수록 영상 처리의 성능이 향상됨을 확인하였다.

제안 시스템은 사용자의 얼굴 정보만을 가지고 사용자가 출현하는 영상 데이터들을 추출하여 데이터셋으로 만들 수 있다. 이는 CCTV 인물 감시 시스템 혹은 개인화 맞춤형 데이터셋 추출 등에 활용될 수 있다. 예를 들면, 특정 사용자의 생활 패턴

및 행동 특징들을 영상 데이터에 기반하여 인공지능 학습을 진행하고 자고자 할 때, 다양한 CCTV나 홈카메라 등을 통하여 해당 사용자만 출현하는 영상 데이터를 수집하기에는 매우 높은 비용이 요구된다. 하지만 제안 시스템은 사용자가 생활하며 실시간으로 수집되는 영상 데이터를 엡지 컴퓨팅 환경에서 효율적인 분석을 통해 저비용으로 추출할 수 있을 것으로 기대할 수 있다.

향후 연구로는 얼굴 클러스터링의 파라미터 최적화를 통해 검색 성능을 추가 개선하고, 제안 시스템을 통해 학습 데이터를 확보하여 시계열 기반의 인공지능 모델을 구축하고자 한다.

References

- [1] W. Riaz, G. Chenqiang, A. Azeem, Saifullah, J. A. Bux, and A. Ullah, "Traffic Anomaly Prediction System Using Predictive Network", *Remote Sensing*, Vol. 14, No. 3, pp. 447, Jan. 2022. <https://doi.org/10.3390/rs14030447>.
- [2] K. J. Cheoi, "Temporal Saliency-Based Suspicious Behavior Pattern Detection", *Applied Sciences*, Vol. 10, No. 3, pp. 1020, Feb. 2020. <https://doi.org/10.3390/app10031020>.
- [3] C. Huang, Z. Wu, J. Wen, Y. Xu, Q. Jiang, and Y. Wang, "Abnormal Event Detection Using Deep Contrastive Learning for Intelligent Video Surveillance System", *IEEE Transactions on Industrial Informatics*, Vol. 18, No. 8, pp. 5171-5179, Aug. 2022. <https://doi.org/10.1109/TII.2021.3122801>.
- [4] M. T. Bhatti, M. G. Khan, M. Aslam, and M. J. Fiaz, "Weapon Detection in Real-Time CCTV Videos Using Deep Learning", *IEEE Access*, Vol. 9, pp. 34366-34382, Feb. 2021. <https://doi.org/10.1109/ACCESS.2021.3059170>.
- [5] A. Nadeem, M. Ashraf, N. Qadeer, K. Rizwan, A. Mehmood, A. AlZahrani, F. Noor, and Q. H. Abbasi, "Tracking Missing Person in Large Crowd Gathering Using Intelligent Video Surveillance",

- Sensors, Vol. 22, No. 14, pp. 5270, Jul. 2022. <https://doi.org/10.3390/s22145270>.
- [6] M. B. Ayed, S. Elkosantini, S. A. Alshaya, and M. Abid, "Suspicious Behavior Recognition Based on Face Features", IEEE Access, Vol. 7, pp. 149952-149958, Oct. 2019. <https://doi.org/10.1109/ACCESS.2019.2947338>.
- [7] J. Zheng, R. Ranjan, C. H. Chen, J. C. Chen, C. D. Castillo, and R. Chellappa, "An Automatic System for Unconstrained Video-Based Face Recognition", IEEE Transactions on Biometrics, Behavior, and Identity Science, Vol. 2, No. 3, pp. 194-209, Jul. 2020. <https://doi.org/10.1109/TBIOM.2020.2973504>.
- [8] E. M. Saoudi and J. A. Said, "A distributed Content-Based Video Retrieval system for large datasets", Journal of Big Data, Vol. 8, No. 1, pp. 1-26, Jun. 2021. <https://doi.org/10.1186/s40537-021-00479-x>.
- [9] A. K. Prabavathy, I. T. Joseph S, M. Mythily, and J. A. M. Rexie, "Object based Video Retrieval with multiple features matching approach", 2022 8th International Conference on Advanced Computing and Communication Systems, Coimbatore, India, Vol. 1, pp. 118-123, Jun. 2022. <https://doi.org/10.1109/ICACCS54159.2022.9785106>.
- [10] Y. Jiachen, J. Bin, and S. Houbing, "A distributed image-retrieval method in multi-camera system of smart city based on cloud computing", Future Generation Computer Systems, Vol. 81, pp. 244-251, Apr. 2018, <https://doi.org/10.1016/j.future.2017.11.015>.
- [11] Y. Hongyang, C. Mengqi, H. Li, and J. Chunfu, "Secure video retrieval using image query on an untrusted cloud", Applied Soft Computing, Vol. 97, pp. 106782, Dec. 2020. <https://doi.org/10.1016/j.asoc.2020.106782>.
- [12] W. J. Kim and C. H. Youn, "Lightweight Online Profiling-Based Configuration Adaptation for Video Analytics System in Edge Computing", IEEE Access, Vol. 8, pp. 116881-116899, Jun. 2020. <https://doi.org/10.1109/ACCESS.2020.3004571>.
- [13] A. A. Ahmed and M. Echi, "Hawk-Eye: An AI-Powered Threat Detector for Intelligent Surveillance Cameras", IEEE Access, Vol. 9, pp. 63283-63293, Apr. 2021. <https://doi.org/10.1109/ACCESS.2021.3074319>.
- [14] D. Lee and D. Ko, "A Study on the Design of Forest IoT Network with Edge Computing", Journal of Korean Institute of Information Technology, Vol. 16, No. 10, pp. 101-109, Oct. 2018. <https://doi.org/10.14801/jkiit.2018.16.10.101>.
- [15] R. Dautov, S. Distefano, D. Bruneo, F. Longo, G. Merlino, and A. Puliafito, "Data Processing in Cyber-Physical-Social Systems Through Edge Computing", IEEE Access, Vol. 6, pp. 29822-29835, May 2018. <https://doi.org/10.1109/ACCESS.2018.2839915>.
- [16] A. Alam, M. N. Khan, J. Khan, and Y. Lee, "IntelliBVR - Intelligent Large-Scale Video Retrieval for Objects and Events Utilizing Distributed Deep-Learning and Semantic Approaches", 2020 IEEE International Conference on Big Data and Smart Computing, Busan, Korea, pp. 28-35, Apr. 2020. <https://doi.org/10.1109/BigComp48618.2020.0-103>.
- [17] J. Chang, C. Kang, J. Yoon, J. Cho, J. Jung, and J. Chun, "Development of Real-time Video Search System Using the Intelligent Object Recognition Technology", The Journal of the Institute of Internet, Broadcasting and Communication, Vol. 20, No. 6, pp. 85-91, Dec. 2020. <https://doi.org/10.7236/JIIBC.2020.20.6.85>.
- [18] D. Pham, V. Vu, B. Yoon, K. Kim, and K. P. Kim, "A Conceptual Architecture for Deep-learning-based Video-Objects Retrieving System", 2022 24th International Conference on Advanced Communication Technology, pp. 234-238, Mar. 2022. <https://doi.org/10.23919/ICACT53585.2022.9728916>.

- [19] H. Sun, Y. Yu, K. Sha, and B. Lou, "mVideo: Edge Computing Based Mobile Video Processing Systems", IEEE Access, Vol. 8, pp. 11615-11623, Dec. 2019. <https://doi.org/10.1109/ACCESS.2019.2963159>.
- [20] W. Dou, X. Zhao, X. Yin, H. Wang, Y. Luo, and L. Qi, "Edge Computing-Enabled Deep Learning for Real-time Video Optimization in IIoT", IEEE Access, Vol. 17, pp. 2842-2851, Apr. 2021. <https://doi.org/10.1109/TII.2020.3020386>.
- [21] S. Wan, S. Ding, and C. Chen, "Edge computing enabled video segmentation for real-time traffic monitoring in internet of vehicles", Pattern Recognition, Vol. 121, pp. 108146, Jan. 2022. <https://doi.org/10.1016/j.patcog.2021.108146>.
- [22] E. Kim, "The Use of Haar Cascade Result selection algorithm to check Wearing Masks and Fever Abnormality", Journal of the Korea Institute of Information and Communication Engineering, Vol. 26, No. 2, pp. 193-198, Feb. 2022. <https://doi.org/10.6109/jkiice.2022.26.2.193>.
- [23] G. Wtewart and M. Al-Khassaweneh, "An Implementation of the HDBSCAN* Clustering Algorithm", Applied Sciences, Vol. 12, No. 5, pp. 2405, Feb. 2022. <https://doi.org/10.3390/app12052405>.

김 장 원 (Jangwon Gim)



2008년 2월 : 고려대학교
컴퓨터학과(이학석사)
2012년 8월 : 고려대학교
컴퓨터·전파통신공학과(공학박사)
2013년 3월 ~ 2017년 3월 :
한국과학기술정보연구원
선임연구원

2017년 4월 ~ 현재 : 군산대학교 소프트웨어학부 부교수
관심 분야 : 자연어 처리, 지식그래프, 지식임베딩,
실시간 빅데이터 분석

이 석 훈 (Sukhoon Lee)



2009년 2월 : 고려대학교
전자및정보공학부(학사)
2011년 2월 : 고려대학교
컴퓨터·전파통신공학과(공학석사)
2016년 2월 : 고려대학교
컴퓨터·전파통신공학과(공학박사)
2016년 3월 ~ 2017년 3월 :

아주대학교 의료정보학과 연구강사
2017년 4월 ~ 현재 : 군산대학교 소프트웨어학부 부교수
관심 분야 : 사물인터넷, 엡지 컴퓨팅, 데이터 품질,
시맨틱웹

저자소개

최 진 수 (Jinsu Choi)



2023년 2월 : 군산대학교
소프트웨어학과(학사)
2023년 2월 ~ 현재 :
(주)아프리카TV 사원
관심분야 : 사물인터넷, 엡지
컴퓨팅