|  |
| --- |
| 얼굴인식 알고리즘 분석 및 소켓통신에서의 활용 |
|  |
| 김현빈\*, 권윤호\*, 이준석\*, 심재창\*  \*안동대학교 컴퓨터공학과  e-mail : hyeon5146@naver.com |
| Face Detect Algorithm Analysis and Application in Socket Communication |
|  |
| HyeonBin Kim\*, Yoon Ho kwon\*, JoonSeock Lee\*, Jaechang Shim\*  \*Dept. of Computer Science, Andong National University |
|  |
| 요  약  인공지능이 많은 분야에서 활용되고 있는 현대 환경에서 인공지능에 대해 알아보고 체험하는 경험은 중요하다. 따라서 인공지능 알고리즘 4가지에 대해 분석 및 속도를 측정해보고 그 결과를 바탕으로 소캣 통신을 통해 클라이언트가 전송한 사진을 서버에서 얼굴을 찾아 리턴하는 실습 프로그램 제작 및 연구를 진행한다. 본 논문에서는 ‘Deepface’,’haar casecade’,’dlib’,’MTCNN’을 핵심적으로 다루고 소켓 통신 예제는 얼굴 인식프로그램에 적합하도록 데이터 전송이 가능하도록 설계한다. |
|  |

**1. 서론**

2022년 한국 기준AI를 알고있거나 들어본 경우는 약99.3%, 그 중 매우 관심있다고 답한 집단이 59.8%이다.[1] 이처럼 현대 사회에서 인공지능에 대한 관심도는 상당수준까지 올라와 있다. 따라서 우리는 인공지능을 직,간접적으로 비전공자에게 인공지능에 대한 새로운 경험을 선사해줄 엔지니어가 되기에 선행하여 인공지능 얼굴인식을 구현하고 이를 소켓통신과 접목하여 두 가지를 동시에 체험할 수 있는 예제를 구현하는 연구를 통해 차세대 핵심 기술에 대한 이해를 쌓고, 연구 결과물로 얼굴인식 알고리즘의 분석과 누구나 체험할 수 있는 실습 프로그램 산출을 목표로 한다.

**2. 관련연구** **(얼굴인식 알고리즘 4가지)**

**2.1 Dlib(HOG 알고리즘)**

**파이썬에서 제공하는 인공지능 알고리즘 라이브러리로 “HOG 알고리즘”은 픽셀 값의 변화를 영상 밝기의 변화와 방향을 기준으로 삼아 얼굴 객체를 추출하는 방식을 사용한다.**

**조사결과 CPU환경에서 4알고리즘 중 상위권의 실행속도를 보였으며, 입력 사진의 사이즈가 일정크기 이상이며 고정 사이즈의 사진에서 더 좋은 성능을 보임. 또한 빛의 Gradient 누적 값을 활용하여 폐색영역에서 우수한 인식률을 보인 특징이 있었지만,**

**최소 사진 크기 80\*80을 요구한다는 단점을 보인다**

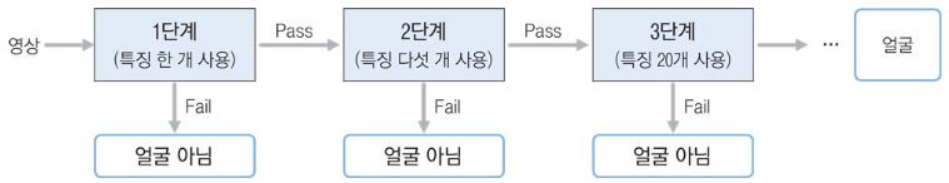
**[2][3].**

**2.2 Haar CaseCade**

**텍스트, 스크린샷, 기기, 주방기기이(가) 표시된 사진

자동 생성된 설명**

**그림1. 하르필터**

****

**그림2. CaseCade모델**

**Open CV에서 기본으로 제공되는 라이브러리로 “하르필터”,”Case Cade”모델의 검사를 접목시킨 것 이다. 하르필터는 얼굴인식에 최적화된 흑과 백색이 된 필터이고, Case Cade 방식은 검사 후 조건에 부합하는 경우에만 다음 검사를 수행할 수 있도록 하여 검사 속도를 향상시키는 방식이다[4].**

**위 알고리즘은 이미지나 영상을 픽셀이 아닌 직사각형 형태의 하르필터를 사용하기여 검사를 진행하여 일반적인 픽셀을 이용한 얼굴인식 방법보다 동작속도가 빠르며, 특히 가변적인 사이즈의 사진에서 더 좋은 성능을 보인다. 하지만 폐색 영역에는 취약하다는 단점이 있었다.**

**2.3 MTCNN**

각각P-net, R-net, O-net으로 구분되는 세 CNN을 차례로 통과하는 CaseCade모델으로 각 영역에서는 다음과 같은 과정을 통해 최종 얼굴을 추출할 수 있다[5][6].

2.3.1 P-net

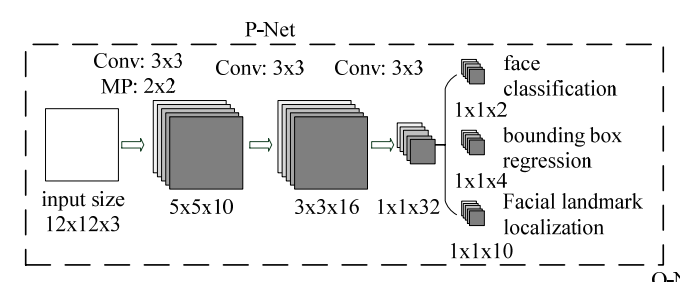


그림3. P-net 과정

원본 입력 이미지에 다수의 얼굴이 포함된 경우를 대비하여 이미지를 작은 조각으로 리사이즈한 리스트를 만든다. 이를 바탕으로 3개의 영역(face classification, bounding box regression, Facial landmark localization)으로 분리된 리턴을 제공하여 다음 단계의 입력이 된다

2.3.2 R-net

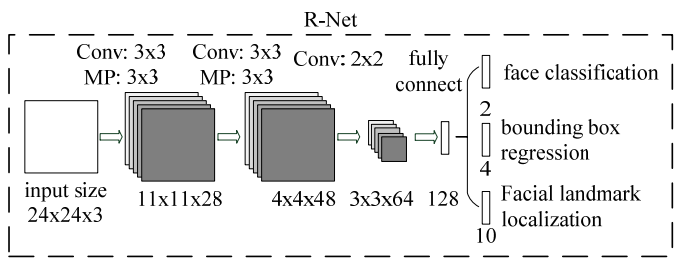
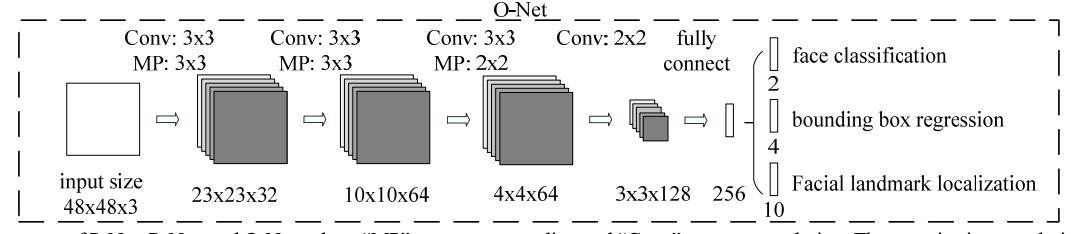


그림4. R-net 과정

P-net을 통해 얼굴로 추정되는 박스들의 리스트를 받아 Fulli Connected Layer 추가를 통해 이 중 진짜 얼굴 부분을 추출하는 과정을 거친다.

2.3.3 O-net

그림 5. O-net 과정

앞서 두 과정을 통해 찾아낸 얼굴 후보 박스를 48\*48크기의 입력으로 받아 점점 필터의 크기를 키우며 더욱 추상적인 정보를 찾아낸다. 이 과정을 통해 산출되는 최종 아웃풋이 탐지된 얼굴이 된다.

**2.4 DeepFace(facebook)**

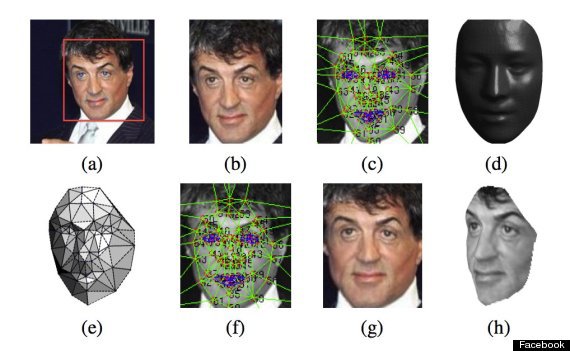


그림6. DeepFace, FaceDetect 과정

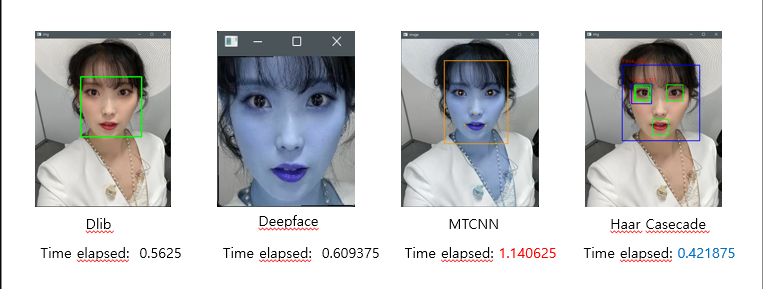
페이스북에서 개발한 얼굴인식 알고리즘으로 페이스북의 방대한 얼굴 데이터를 통해 학습된 알고리즘으로 정확도가 97.25%로 매우 우수하다는 특징이 있다[7].

그림[6]의 과정을 통해 딥페이스의 알고리즘을 파악할 수 있다. 우선 6개의 특징점을 잡는다.(a)

이를 통해 2D로 정렬된 자료를 얻고(b), 2D자료와 67개의 기준점, 그리고 그에 상응하는 Delaunay 삼각 측량법과 등고선 삼각형을 통해 삼각형들의 불연속을 피한다(c). 위 과정을 거치면 3D자료가 2D자료로 변경된다(d). 어두운 부분은 얼굴의 사각지대를 표현한 모습이다(e). 아핀 변환이 적용된 조각들을 이용하여 3D모델로부터 67개의 특징적인 점들이 유도된다(f). 최종 추출된 정면의 모습(g). 새롭게 유도된 얼굴의 측면 예상도(h).

MTCNN방식은 여러 사이즈의 필터를 거치는 과정을 통해 정확도가 매우 우수하며. 폐색 영역에 강하고 CPU환경에서는 4 알고리즘 중 가장 느린 속도를 보인다. 이미지의 크기가 커질수록 CNN interface의 가 수가 급격히 증가하기 때문이다. (실제로 1080\*720의 고 해상도 이미지를 입력하면 GPU를 사용하여도 10초 이상의 시간이 소요되었음.)

**3.얼굴인식 알고리즘 속도분석 결과**

****

**그림.7 4알고리즘 속도 비교(ms)**

**(Window 11, Intel® Core(TM) i5-10210U CPU @1.60GHz 2.11GHz, 16GB RAM)**

**본 테스트 결과 Haar CaseCade가 가장 빠른 처리속도(0.141875)를 보였고, MTCNN이 가장 느린 처리속도(1.140625)를 보였다. 본 결과는 테스트마다 순위가 경미하게 변동되어 총 10회의 같은 이미지를 기준으로 테스트를 진행한 후 그 평균값을 사용하여 부여한 순위이다. 따라서 본 논문는 Haar CaseCaded알고리즘을 적용하여 실습 예제를 제작할 것 이다.**

**4. 소켓통신원리**

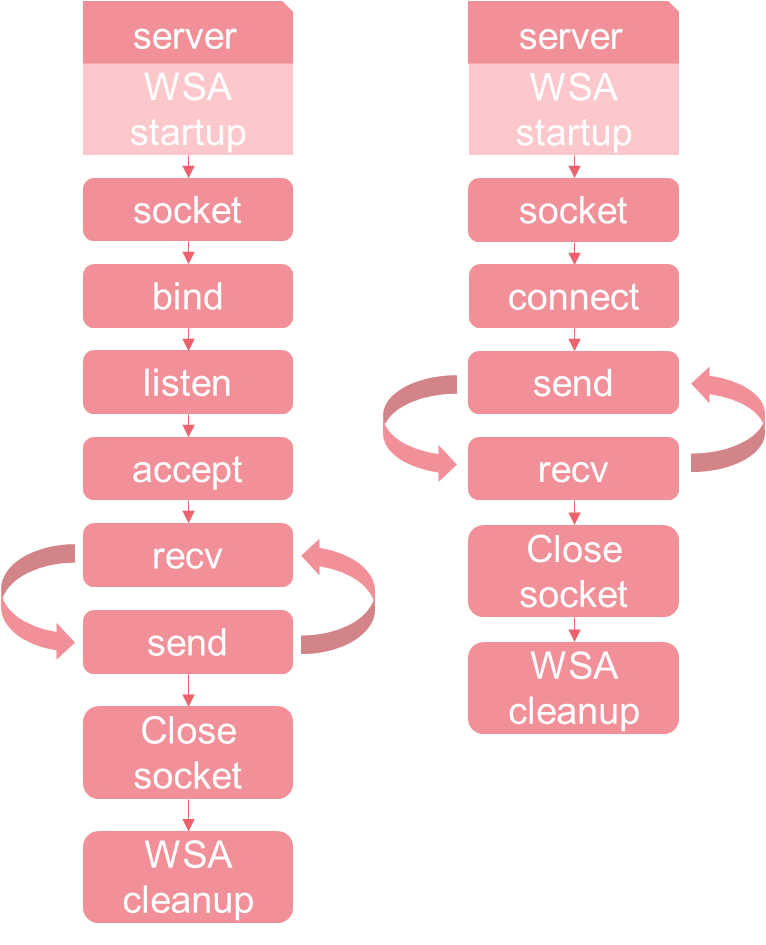
****

그림8.동기식 채팅 프로그램 알고리즘

socket(네트워크 주소 체계, 소켓 타입, 프로토콜)

소켓을 생성하는 함수이며 위와 같은 매개변수를 가지고 초기화 한다. 소켓 생성을 실패하면 -1을 반환한다[8][9].

• 네트워크주소체계:

IPv4(AF\_INET), IPv6(AF\_INET6)

•소켓타입:

TCP(SOCK\_STREAM), UDP(SOCK\_DGRAM)

• 프로토콜:

TCP(IPPROPTO\_TCP), UDP(IPPROTO\_UDP)

bind(소켓 변수, 서버 주소 구조체, 서버 주소 구조체의 크기)

서버 측의 소켓에 IP와 포트를 할당하여 네트워크 인터페이스와 묶일 수 있도록 합니다.

• IPv4 소켓 주소 구조체(SOCKADDR\_IN): 소켓 주소체계(sin\_family), 포트(sin\_port), IPv4 주소(sin\_addr)

실제로 소켓 주소 구조체를 이용할 때 IPv4 소켓 주소 구조체를 일반 소켓 구조체(SOCKADDR)로 변환하여 사용해야 한다.

listen(소켓 변수, 백 로그 큐의 크기)

클라이언트로부터 연결 요청을 기다린다. 백 로그란 동시에 연결을 시도하는 최대 클라이언트 수를 의미한다.

connect(소켓 변수, 서버 주소 구조체, 서버 주소 구조체 크기)

클라이언트 측에서 서버에 연결을 요청한다. 소켓 생성에 실패하는 경우 -1을 반환한다.

• IPv4 소켓 주소 구조체: 소켓 주소체계(sin\_family), 포트(sin\_port), IPv4 주소(sin\_addr)

실제로 소켓 주소 구조체를 이용할 때는 IPv4 소켓 주소 구조체를 일반 소켓 구조체(SOCKADDR)로 변환하여 사용해야 한다.

accept(소켓 변수, 클라이언트 주소 구조체 변수, 클라이언트 주소 구조체 크기)

서버 측에서 클라이언트의 연결을 수락한다. accept() 함수 내부에서 클라이언트 주소를 설정한 뒤에 통신에 사용할 클라이언트의 소켓을 반환한다. 오류가 발생하는 경우 -1을 반환한다.

실제로 소켓 주소 구조체를 이용할 때는 IPv4 소켓 주소 구조체를 일반 소켓 구조체(SOCKADDR)로 반환하여 사용해야 한다.

send(소켓 변수, 문자열 버퍼, 문자열 버퍼 크기, 플래그)

상대방에게 데이터를 보낸다. 특정한 소켓으로 문자열 버퍼에 담긴 내용을 전송한다. 플래그는 특별한 경우가 아니라면 0을 설정한다. 데이터 전송에 실패하는 경우 -1을 반환한다.

recv(소켓 변수, 문자열 버퍼, 문자열 버퍼 크기, 플래그)

데이터를 수신하여 특정 문자열 버퍼에 담는다. 플래그는 특별한 경우가 아니라면 0을 설정하고 데이터 수신에 실패하는 경우 -1을 반환한다.

closesocket(소켓 변수)

열린 소켓을 닫는다.

**5. 소켓통신(MFC) 에서의FaceDetect활용.**

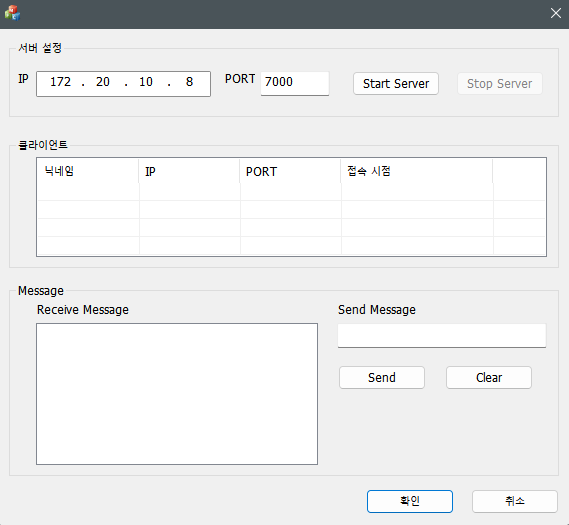


그림9. 소켓통신 (서버)

텍스트이(가) 표시된 사진

자동 생성된 설명

그림10.소켓통신(클라이언트)

본 논문의 연구과정에서는 알고리즘을 분석한 결과를 토대로 오픈소스 MFC소켓통신 예제를 결합하여 클라이언트에서 사진 데이터를 서버로 보내면 서버에서 우리 팀원들이 선택한 최적의 알고리즘을 적용하여 원본 사진에 박스를 통해 얼굴부분을 표시한 후 다시 클라이언트에게 산출물을 리턴한다.

[그림7]의 화면에서 IP와 포트 번호를 입력하여 서버를 개설하고, 클라이언트 측에서도 같은 IP와 포트를 통해 서버에 접근할 수 있다.

본 서버-클라이언트 프로그램은 소켓 통신을 MFC를 활용하여 구현한 것으로 간단한 채팅과 데이터를 서버로 전송 가능하여 여러 클라이언트들 간의 채팅과 데이터 교류가 가능하다.

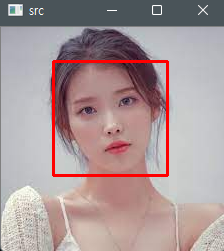


그림 11. 클라이언트로부터 받은 사진에서 추출한 얼굴 (서버의 리턴값)

**5. 결론 및 향후 연구**

본 논문에서는 현대의 인공지능에 대한 관심도에 답하는 얼굴인식 알고리즘 분석 및 소켓통신에서의 활용 예시를 통해 비 전공자들에게 안면인식을 체험할 수 있는 실습 프로그램을 제작하였다.

결론적으로 본 논문에서는 단순히 같은 하드웨어 환경에서의 알고리즘의 실행속도를 기준으로 분석하여 “Haar CaseCade”알고리즘이 채택되었지만, 이것이 정답 이라고는 할 수 없다. 따라서 상황에 맞는 적절한 알고리즘 선택이 중요하다. 이는 추후 연구과정으로 남겨두며 해결해야할 부분이다.

소켓 통신의 경우 실생활에서 가장 흔하고 다루기 편리한 기술이라 인공지능과 엮어 함께 제공하면 비전공자들의 실습에 흥미를 돋우는 효과를 기대한다.

**참고문헌**

////

[1]Edaily(2022)<https://www.edaily.co.kr/news/read?mediaCodeNo=257&newsId=01079126629175856> (accessed July 10, 2022)

[2] Lib C++ Library. <http://dlib.net> (accessed May 08, 2022)

[3] 콤퓨타 사이언스(dlib를 이용한 얼굴탐색) <https://hayunjong83.tistory.com/38> (july 07, 2020)

[4] 하르 분류기를 이용한 얼굴 검출 <https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=infoefficien&logNo=220925934179> (accessed February 03, 2017)

[5]갈아먹는 머신러닝<https://yeomko.tistory.com/16> (accessed January 06, 2020)

[6]Y.S.Ha 2017 *AutumnAcademic Announcement 2017.*

[7]이미지 인식-페이스북(딥페이스) <https://blog.daum.net/md114/8932546> (accessed April 10, 2019)

[8]곰돌이놀이터-소켓(SOCKET)통신이란? <https://helloworld-88.tistory.com/215> (accessed October , 2019)

[9]C++를 이용한 TCP소켓통신 구현 <https://kevinthegrey.tistory.com/m/26> (accessed Feburary 27, 2018)