



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2020-0087892  
(43) 공개일자 2020년07월22일

- |                                                                                                                                                                                                                                                                                                                                             |                                                                                                                                                                                                                                                                      |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>(51) 국제특허분류(Int. Cl.)<br/> <b>G06K 9/00</b> (2006.01) <b>G06N 20/00</b> (2019.01)<br/> <b>G06N 3/08</b> (2006.01) <b>H04N 7/18</b> (2006.01)</p> <p>(52) CPC특허분류<br/> <b>G06K 9/00711</b> (2013.01)<br/> <b>G06N 20/00</b> (2019.01)</p> <p>(21) 출원번호 <b>10-2018-0172481</b><br/> (22) 출원일자 <b>2018년12월28일</b><br/> 심사청구일자 <b>없음</b></p> | <p>(71) 출원인<br/> <b>한국기술교육대학교 산학협력단</b><br/> 충청남도 천안시 동남구 병천면 충절로 1600 (한국기술교육대학교내)</p> <p>(72) 발명자<br/> <b>성재용</b><br/> 서울특별시 성동구 왕십리로 80 11동 805호 (성수동1가, 동아아파트)<br/> <b>서희석</b><br/> 경기도 의왕시 내손중앙로 46-13, B동 202호 (내손동, 경춘빌라)</p> <p>(74) 대리인<br/> <b>김진동</b></p> |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

전체 청구항 수 : 총 2 항

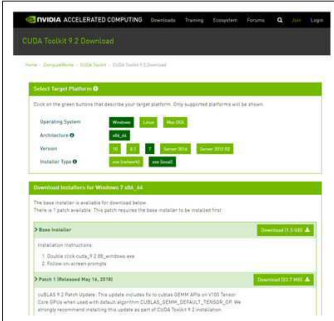
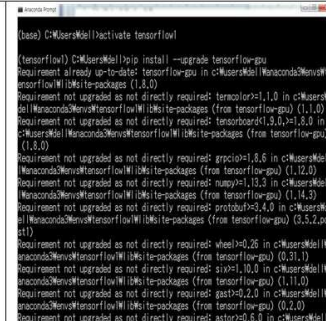
(54) 발명의 명칭 딥러닝 기반 영상 내 객체 인식 시스템

(57) 요약

본 발명은 딥러닝 기반 영상 내 객체 인식 시스템에 관한 것으로, 객체 머신러닝을 통한 딥러닝 기반의 영상 내 자동 객체 인식/분석 시스템을 개발한다.

학습자 정보, 문제정보, 동영상 등 미용 관련 모든 자료에 대한 최초의 빅데이터 구성을 통한 '이/미용 스마트 이터닝 플랫폼' 구축과 이를 기반으로 한 취약 부분 분석 도출 알고리즘 구현 분야별 표준적 동영상 이미지 커트 대비 학습자의 실습 동영상 이미지의 자동 비교 분석 및 적합률 데이터를 제공해주는 딥러닝 알고리즘 적용 시스템 구현

대 표 도 - 도1

|                                                                                                              |                                                                                                       |                                                                                                                 |
|--------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|
|  <p>① CUDA Toolkit 설치</p> |  <p>② cuDNN 설치</p> |  <p>③ Tensorflow-GPU 설치</p> |
|--------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------|

(52) CPC특허분류

*G06N 3/08* (2013.01)

*H04N 7/18* (2013.01)

---

## 명세서

### 청구범위

#### 청구항 1

객체 탐지 디렉토리 및 환경변수 구성

사진 수집 및 레이블 지정

학습 자료 생성

학습구성

학습

추론그래프 생성

객체 인식 분류기를 통하여 학습된 모델 적용

TensorFlow 머신러닝/딥러닝 기반 CCTV 자동 객체 인식 시스템.

#### 청구항 2

분석 모듈: 다중 객체들의 빠르고 정확한 결과를 도출하기 위한 Fast R-CNN 알고리즘 적용

머신러닝/딥러닝: TensorFlow를 이용해 데이터 머신러닝 수행

학습자 정보, 문제정보, 동영상 등 미용 관련 모든 자료에 대한 최초의 빅데이터 구성을 통한 '이/미용 스마트 이터닝 플랫폼' 구축과 이를 기반으로 한 취약 부분 분석 도출 알고리즘 구현 분야별 표준적 동영상 이미지 커트 대비 학습자의 실습 동영상 이미지의 자동 비교 분석 및 적합률 데이터를 제공해주는 딥러닝 알고리즘 적용 시스템 구현

k means 알고리즘은 k 개의 평균(means) 벡터를 이용한 클러스터링(군집화) 알고리즘이며 데이터들의 분산을 최소화하는 k개의 평균 벡터를 구한다.

1. (처음 중심 값 선택) 랜덤하게 초기 중심 값(centroid)을 선택한다.
2. (클러스터 할당) k 개의 중심 값과 각 개별 데이터간의 거리(distance)를 측정한다. 가장 가까운 클러스터에 해당 데이터를 할당(assign)한다.
3. (새 중심 값 선택) 클러스터마다 새로운 중심 값을 계산한다.
4. (범위 확인-convergence) 선택된 중심 값이 변화가 어느 정도 없다면 멈춘다. 만약 계속 변화가 있다면 1 번 부터 반복한다.

decision tree 알고리즘 어떤 항목에 대한 관측값과 목표값을 연결시켜주는 예측 모델 로써 결정 트리를 사용.

1. 최초의 decision 생성한다.
2. 하위 집합은 최초의 decision 노드의 branches가 된다.
3. 만약 branch에 있는 데이터가 모두 같은 분류 항목에 속하면 제대로 분류가 된 것이며 생성을 종료한다.
4. 그렇지 않다면, 이 하위 집합에서는 다시 분할 절차를 반복한다.
5. 모든 데이터가 분류될 때까지 이 절차를 반복한다.

객체 인식 시스템.

### 발명의 설명

### 기술 분야

[0001] 본 발명은 딥러닝 기반 영상 내 객체 인식 시스템에 관한 것이다.

## 배경 기술

[0002] 객체 머신러닝을 통한 딥러닝 기반의 영상 내 자동 객체 인식/분석 시스템을 개발한다.

[0003] 현재 국내 미용분야의 이러닝 학습시스템은 아직까지 종합적인 체계로 구현되어 있지 않으며, 특히 미용사 국가 기술시험의 이론 및 실기시험 과목에 대해서는 감점 및 불합격 요인에 대한 개인별 맞춤형 보완 지원시스템 구현이 없다.

[0004] 헤어 컷트 분야의 예를 들면 컷트 실기의 종류는 이사도라, 스파니엘, 그라듀에이션, 레이어 드가 있는데, 불합격 사유의 예를 들면 이사도라의 경우 "앞머리가 뒤통보다 짧아지는 스타 일로 마네킹의 머리가 15초 숙여야 하는 컷트로, 통상에서 턱선과 맞추어야 하는데 양쪽이 잘 맞지 않는 부분이 생기고 층이 생기면……" 이다.

[0005] 따라서 이와 같은 부분의 취약점 분석과 보완 서비스가 절대적으로 필요한 부분이며, 차후 예는 디자이너로서의 성장을 위한 심화학습까지 제공할 예정으로, 이것이 본 과제 시스템의 개발 의도이다.

[0006] 미용 분야는 크게 헤어, 피부, 네일, 메이크업 분야로 대분류되며, 본 과제 제안서는 미용 분야 전 분야인 4개 분야에 대한 이론 부분을 적용할 것이며, 실습 부문에 있어서는 우선 헤어 분야 중 컷트 분야를 적용시킬 것이다.

[0007] 당사가 검토 조사한 바에 따르면 문제점 현상이 크게 2가지 이다.

[0008] 첫째는 학습자들에 대한 모의고사 평가결과가 학습자 교육수준이라든가, 기존 직업 그리고 개인 호불호 등에 따라 천차만별로 나타나기 때문에, 안타깝게도 학원에서는 많은 학습자 개개인 별로 강습을 할 수가 없고 정해진 과정에 따라 교육을 진행하기 때문에 학습자 취약점에 대한 맞춤형 교육이 결코 이루어질 수가 없다는 것이다.

[0009] 둘째는 이와 같은 사유로 학습자의 시험 합격률은 통상적으로 30% 정도의 매우 저조한 실정이고, 적어도 2~4 회차가 되어서야 합격하는 실정이다.

[0010] 따라서 학습자가 최고의 기술력 습득을 통하여 최단시간에 목적인 학습 목표를 달성 시키기 위한 아래와 같은 시스템 구현 지원이 절대적으로 필요하게 되었다.

[0011] 미용 분야별 /숙련 단계별 학습 과정에 대한 체계적인 통합 스마트 이러닝 시스템 구축 필요

[0012] 학습자 개인별 취약 분야 (이론, 실기)에 대하여 자동적으로 분석, 선정 제시해줄 수 있는 지능형 분석 알고리즘 (빅데이터 분석, 딥러닝 분석) 개발 적용 시스템 필요

[0013] 최초의 Global Version "이/미용 스마트 통합 학습 솔루션" 선도적 모델 기반 구축 필요

[0014] 특히 이와 같은 '지능형 스마트 이/미용 시스템'에 대한 구현의 필요성에 대하여는 '대한미용사회중앙회'사무부 총장님이나, 청주'보떼아트쿨'원장이며, 현재 서원대학교 뷰티학과 교수이신'한희선'님으로부터 직접적으로 강력한 필요성 요청을 받은 상황이다.

## 발명의 내용

### 해결하려는 과제

[0015] 본 발명은 객체 머신러닝을 통한 딥러닝 기반의 영상 내 자동 객체 인식/분석 시스템을 개발한다.

[0016] 기존은 인터넷을 통하여 주요 강사 또는 개인별 취득 정보 등록에 의한 학습자 개인의 임 의적 검색을 통하여 단순한 정보만을 습득하였다.

[0017] 신규 제시 시스템은 미용분야에 대하여 수준별, 계층별, 단계별 정보(이론, 사진, 동영상)를 종합 관리함으로써, 개인별 학습자에 맞는 체계적인 맞춤형 학습 시스템으로 구현되며, 이러한 종합된 정보에 대한 빅데이터 구성을 통하여 지능형 분석 알고리즘을 적용한 개인별 취약부분을 분석 제공해 주는 최초의 빅데이터, 딥러닝 IT기술 시스템을 구현할 계획이다.

### 과제의 해결 수단

[0018] 본 발명의 한 실시예에 따른 딥러닝 기반 영상 내 객체 인식 시스템은, TensorFlow 머신러닝/딥러닝 기반 CCTV

자동 객체 인식 시스템 개발로서, 분석 모듈: 다중 객체들의 빠르고 정확한 결과를 도출하기 위한 Fast R-CNN 알고리즘 적용하고, 머신러닝/딥러닝: TensorFlow를 이용해 데이터 머신러닝 수행한다.

[0019] 학습자 정보, 문제정보, 동영상 등 미용 관련 모든 자료에 대한 최초의 빅데이터 구성을 통한 '이/미용 스마트 이터닝 플랫폼' 구축과 이를 기반으로 한 취약 부분 분석 도출 알고리즘 구현 분야별 표준적 동영상 이미지 커트 대비 학습자의 실습 동영상 이미지의 자동 비교 분석 및 적합률 데이터를 제공해주는 딥러닝 알고리즘 적용 기술 구현

### 발명의 효과

[0020] 본 발명의 실시예에 따르면, 사용자 별 커스터마이징이 가능하고 확장성이 우수한 솔루션으로서, 객체 머신러닝을 통한 자동 객체인식 딥러닝 솔루션 개발과 원하는 객체를 자유롭게 Train 시켜 영상 내에서 검출 가능하다.

[0021] 본 발명의 실시예에 따르면, CCTV 내 객체 인식 시스템으로 침입탐지, 사람계수 등 보안/마케팅 분야 활용 가능하다. 사회 전반적으로 안전, 보안의식이 향상됨에 따라 보안/방법분야를 중심으로 교통/재해감시 등의 공공분야와 기업 내 보안 및 마케팅, 고객관리 등의 민간분야 등으로 활용용도와 범위 확장 용이하다. 그리고 상태감지(Status Detection)을 통해 사람이나 물체의 움직임과 더불어 관제 구역에 대한 상태 감지 기능을 추가하여 부가적인 보안기능이 요구되는 환경에서 활용 가능 CCTV로 촬영되는 관제구역에서의 변화에 대해 감지가 가능하다. 또한, 특정 물체의 움직임 또는 사라짐을 감지하여 특정분야 활용 가능하다.

### 도면의 간단한 설명

- [0022] 도 1은 본 발명의 한 실시예에 따른 TensorFlow-GPU 구성
- 도 2는 객체 탐지 디렉토리 및 환경변수 구성
- 도 3은 사진(이미지) 수집
- 도 4는 학습 이미지와 테스트 이미지 데이터에 대한 레이블 지정
- 도 5는 학습 자료 생성
- 도 6은 학습구성
- 도 7은 학습
- 도 8은 추론그래프 생성
- 도 9는 객체 인식 분류기를 통하여 학습된 모델 적용
- 도 10은 CCTV로 촬영되는 관제구역에서의 변화에 대해 감지
- 도 11은 시스템 활용 가능 분야
- 도 12는 목표 시스템 구성도

### 발명을 실시하기 위한 구체적인 내용

[0023] 이하, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 본 발명의 실시예에 대하여 첨부한 도면을 참고로 하여 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 명세서 전체를 통하여 유사한 부분에 대해서는 동일한 도면 부호를 붙였다.

[0024] 그러면 본 발명의 한 실시예에 따른 딥러닝 기반 영상 내 객체 인식 시스템에 대하여 도 내지 도 을 참고하여 설명한다.

[0025] 도 1 내지 도 12를 참고하면, 본 실시예에 따른 딥러닝 기반 영상 내 객체 인식 시스템은

[0026] AI(머신러닝, 딥러닝 분야) 전문 인력 양성 가능

[0027] TensorFlow 엔진 전문 인력 양성 가능

[0028] 신경망 분야 전문 인력 양성 가능

- [0029] 데이터 분석 알고리즘 개발(신경망, CNN, RNN 등)
- [0030] 객체 인식 딥러닝 개발을 위한 머신러닝
- [0031] TensorFlow 기반 머신러닝, 딥러닝 과정 학습
- [0032] 신경망 구성 내용 학습
- [0033] Python을 이용한 인공지능 개발 과정 학습
- [0034] 객체 인식 딥러닝 시스템으로 커스터마이징이 용이함
- [0035] 인공지능 이미지 색출 분야 기술 확보
- [0036] Python 기반 인공지능 개발이 가능한 초급기술자 양성
- [0037] 4차 산업혁명 핵심인재 양성
- [0038] TensorFlow 머신러닝/딥러닝 기반 CCTV 자동 객체 인식 시스템 개발한다. 시스템 최종 목표구현을 위한 시스템 개발내용으로, 분석 모듈: 다중 객체들의 빠르고 정확한 결과를 도출하기 위한 Fast R-CNN 알고리즘 적용한다. 머신러닝/딥러닝: TensorFlow를 이용해 데이터 머신러닝 수행한다.
- [0039] TensorFlow-GPU 구성
- [0040] ① 객체 탐지 디렉토리 및 환경변수 구성
- [0041] ① 디렉토리 구조 설정: train data와 test data로 분류하여 특정 디렉토리 구조 적용
- [0042] ② 환경변수 구성: 지정한 디렉토리 환경변수 적용
- [0043] ② 사진 수집 및 레이블 지정
- [0044] ② 다양한 배경과 조명 조건을 가진 임의의 대상물체 사진(이미지)수집
- [0045] 대상의 물체가 가려져 있는 이미지, 겹쳐 있는 이미지, 중간 부분만 있는 이미지 등
- [0046] 인식률을 극대화시키기 위해 수백 개의 물체 이미지를 기계에 학습시키고, 다른 임의의 물체가 있는 환경과 다양한 배경, 조도 조건을 갖춘 이미지를 학습시킨다.
- [0047] ③ 학습 이미지와 테스트 이미지 데이터에 대한 레이블 지정
- [0048] ③ 학습 자료 생성
- [0049] 지정한 레이블을 Tensorflow Record를 생성하기 위해 레이블 소스 작업을 하고,
- [0050] Train record와 Test record를 생성하여 Training 준비 자료들을 구성.
- [0051] ④ 학습구성
- [0052] 파이프 라인 구성
- [0053] [어떤 모델, 어떤 매개 변수가 학습에 사용될지 정의]
- [0054] 학습
- [0055] ④ 학습을 하는 동안 TensorBoard를 통해 학습 및 작업의 프로세스 확인
- [0056] mAP(평균 정밀도), loss(손실) 등
- [0057] ⑤ 추론그래프 생성
- [0058] ⑤ 학습 된 모델을 단일 Tensorflow 그래프로 생성
- [0059] 학습이 완료되었을 마지막 단계에 고정 유추 그래프를 생성
- [0060] ⑥ 객체 인식 분류기를 통하여 학습된 모델 적용
- [0061] ⑥ Python 스크립트로 객체 인식 분류 로직[웹캠, 동영상, 이미지] 구현

- [0062] 오픈소스를 기반으로 한 객체 삽입/삭제 및 소스 수정 용이:
- [0063] 오픈소스를 이용한 파이썬(Python)과 TensorFlow 라이브러리를 이용해 구성된 솔루션으로 기존 시장 제품은 중앙 집중형 구성 방식으로 객체 추가/삭제와 소스 수정이 번거로운 반면 본 개발에서는 각 객체를 캡슐화(Encapsulation)시켜 추가/삭제가 용이할 뿐만 아니라 발생하는 오류에 대한 유지보수 측면에서도 우수함을 나타냄.
- [0064] 이/미용 학습을 위한 빅데이터 분석 플랫폼 구축을 통하여 학습자 취약부분을 자동 분석, 제공하는 지능형 이력 시스템 모듈 구현
- [0065] 본 모듈에서는 학습자의 학습 성적 및 평가 기반 다양한 비교 패턴의 Big Data 구성과 학습자의 학습 평가 결과를 바탕으로 함
- [0066] 학습자 과목별 또는 부문별 평가 성적에 대하여는 K-means 방법론을 통한 클러스터링 군집 분석으로 각 군집 별 특성을 분석하고, 개인별 적성 등과 같은 학습자 별 학습 요인을 적용한 decision tree 기법으로 취약점 분석(library개발)
- [0067] 이론 분야는 헤어, 피부, 네일, 메이크업의 전 분야에 대하여, 실습 분야는 우선 헤어 컷트 분야에 대하여 학습자 취약분야 분석 후, 동영상 강의를 제공
- [0068] 추후에는 학습자 취약분야 분석 후 동영상 강의를 헤어, 피부, 네일, 메이크업의 전 분야로 확대할 것이며, 또한 심화학습 과정 동영상 서비스도 같이 제공함.
- [0069] 강의 동영상은 Microsoft Azure Cloud 서비스 이용 구축함
- [0070] 전체 개발 내용

표 1

|                                  |                |      |
|----------------------------------|----------------|------|
| 기본정보 및<br>점수 입력<br>페이지<br>(html) | 개인정보           |      |
|                                  | 출결사항           |      |
|                                  | 단계 선택          |      |
|                                  | 필기<br>실기<br>점수 | 헤어   |
|                                  |                | 메이크업 |
|                                  |                | 피부   |
|                                  |                | 네일   |

[0071]

표 2

|                     |                                                                       |
|---------------------|-----------------------------------------------------------------------|
| 분석<br>페이지<br>(html) | 헤어, 메이크업<br>피부, 네일 등의<br>이론 및 실기 분<br>야의 취약 과목<br>결과 분석 및 결<br>과 표시화면 |
|---------------------|-----------------------------------------------------------------------|

[0072]

표 3

|                                |                                                                              |
|--------------------------------|------------------------------------------------------------------------------|
| 취약과목<br>자동 선정<br>페이지<br>(html) | 헤어, 메이크업, 피<br>부, 네일 등의 이론<br>및 실기 분야의 취<br>약 과목 자동제시<br>결과 화면(학습지<br>및 동영상) |
|--------------------------------|------------------------------------------------------------------------------|

[0073]

[0074]

Big Data 구성과 Data Base 구축

[0075]

구현 내용: 특정 대상 모집단의 수강자 정보 및 학습 data를 이용하여 오픈소스를 기반으로 한 Big Data를 구축하며, 향후 상용솔루션의 기반이 됨

[0076]

구현 방법: 데이터 파악, 이해 → 시나리오 구성 → 분석 플랫폼 구축 → 시나리오 개발

[0077]

이/미용 학습을 위한 빅데이터 분석 플랫폼 구축 및 지능형 분석 알고리즘 적용 학습자 취약부분 자동분석 시스템 개발

[0078]

K-means 이용한 취약점 분석 (학습자 평가 데이터)

[0079]

k means 알고리즘은 k 개의 평균(means) 벡터를 이용한 클러스터링(군집화) 알고리즘 이며 데이터들의 분산을 최소화하는 k개의 평균 벡터를 구한다.

[0080]

1. (처음 중심 값 선택) 랜덤하게 초기 중심 값(centroid)을 선택한다.

[0081]

2. (클러스터 할당) k 개의 중심 값과 각 개별 데이터간의 거리(distance)를 측정한다. 가장 가까운 클러스터에 해당 데이터를 할당(assign)한다.

[0082]

3. (새 중심 값 선택) 클러스터마다 새로운 중심 값을 계산한다.

[0083]

4. (범위 확인-convergence) 선택된 중심 값이 변화가 어느 정도 없다면 멈춘다. 만약 계속 변화가 있다면 1 번 부터 반복한다.

[0084]

Decision Tree 분석 기능을 이용한 취약점 분석 (학습자 평가 요인)

[0085]

decision tree 알고리즘 어떤 항목에 대한 관측값과 목표값을 연결시켜주는 예측 모델 로써 결정 트리를 사용

[0086]

1. 최초의 decision 생성한다.

[0087]

2. 하위 집합은 최초의 decision 노드의 branches가 된다.

[0088]

3. 만약 branch에 있는 데이터가 모두 같은 분류 항목에 속하면 제대로 분류가 된 것이며 생성을 종료한다.

[0089]

4. 그렇지 않다면, 이 하위 집합에서는 다시 분할 절차를 반복한다.

[0090]

5. 모든 데이터가 분류될 때까지 이 절차를 반복한다.

[0091]

공동개발기관 개발내용

[0092]

시큐어 코딩: 개발 단계에서 발생할 수 있는 코드 수준의 취약점을 방지하는 기술지원 서비스 수준 보안: LMS 및 콘텐츠 서버의 보안 기술 지원(OWASP top 10 취약점 제거)

[0093]

네트워크 수준 보안: 콘텐츠 서버와 사용자 단말 간의 안전한 통신을 위한 모듈 지원

[0094]

제품화 계획

[0095]

대한미용사회중앙회, 한국피부미용사회중앙회 등과 같은 주용 한국 이/미용협회 및 주요 전문 학원과의 시스템 구축 사업 협약을 통하여 미용분야 Smart e-Learning의 구현 및 제품화

[0096]

산학 개발 TFT(전문개발자, 교수, 학생 등)를 구성하여 본 사업과제를 구현하여, 필요시에는 특정 연구단체를

연계 활용한다.

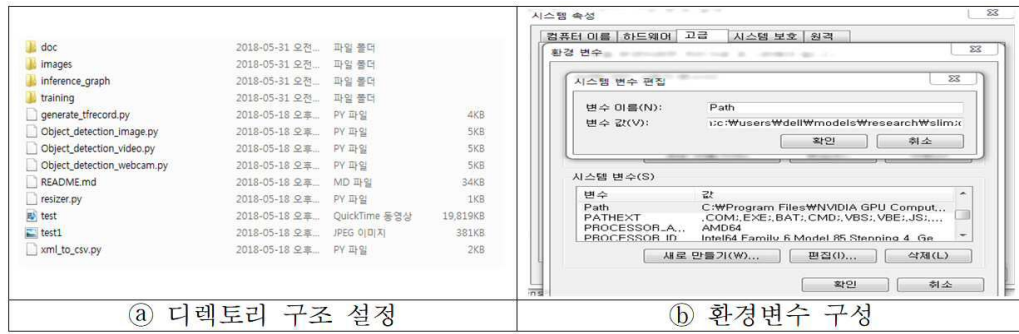
- [0097] 수행 모듈 구성 체제로 개발 추진
- [0098] 분석, 설계 → 개발 구현 → 시스템 테스트, 보완 → 현장 적용 테스트, 보완 → 개발 완료의 단계로 구현
- [0099] 시스템 신뢰성 확보를 위해 공공시험기관 및 별도의 전문 검증단 구성 및 검증 실시
- [0100] 양산 계획
- [0101] 이/미용분야 Smart e-Learning system의 모듈화 (Package Solution) 구현
- [0102] 전문 솔루션 모델 인증을 위해 솔루션 모듈 공식 인증기관의 시스템 인증 획득
- [0103] 단계별 추가 개발 모듈(3~5개)을 위해 인증 완료 제품을 기준으로 하며 과업 추진완료 후 지속적 연계 사업으로 확대 추진
- [0104] (현 과업모델 (1단계)→ 이/미용 4개분야 이론/실습 모델(2단계) → Global Version (3단계))
- [0105] 판로개척 방안
- [0106] 1단계: 한국 이/미용협회 및 주요 전문학원과의 사전 협약을 기반으로한 지역기반 선 적용
- [0107] 충청도 주요 학원 20여개 학원 (1차) → 전국 주요 체인 학원위주 확대 적용(2차)
- [0108] 2단계: On-line 광고 및 Off-line 광고 매체를 통한 전방위 사업화 추진
- [0109] 3단계: 베트남, 중국 등에 대한 박람회 및 대회 참여 사업화 (각 국가별이/미용 협회 협력)
- [0110] 이상에서 본 발명의 바람직한 실시예에 대하여 상세하게 설명하였지만 본 발명의 권리범위는 이에 한정되는 것은 아니고 다음의 청구범위에서 정의하고 있는 본 발명의 기본 개념을 이용한 당업자의 여러 변형 및 개량 형태 또한 본 발명의 권리범위에 속하는 것이다.

## 도면

### 도면1



## 도면2



## 도면3



#### 도면4



#### 도면5

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre>import pandas as pd import tensorflow as tf  from PIL import Image from object_detection.utils import dataset_util from collections import namedtuple, OrderedDict  flags = tf.app.flags flags.DEFINE_string('csv_input', '', 'Path to the CSV input') flags.DEFINE_string('image_dir', '', 'Path to the image directory') flags.DEFINE_string('output_path', '', 'Path to output TFRecord') FLAGS = flags.FLAGS  # TO-DO replace this with label map def class_text_to_int(row_label):     if row_label == 'person':         return 1     else:         return None  def split(df, group):     data = namedtuple('data', ['filename', 'object'])     gb = df.groupby(group)     return [data(filename, gb.get_group(x)) for filename, x in zip(gb.groups.keys(), gb.groups)]  def create_tf_example(filename, path):</pre> | <pre>(base) C:\Users\del\&gt;activate tensorflow (tensorflow) C:\Users\del\&gt;cd models (tensorflow) C:\Users\del\models&gt;cd research (tensorflow) C:\Users\del\models\research&gt;cd object_detection (tensorflow) C:\Users\del\models\research&gt;python generate_tfrecord.py --csv_input=images\train_labels.csv --image_dir=images\train --output_path=train.record Successfully created the TFRecords: C:\Users\del\models\research\object_detection\train.record (tensorflow) C:\Users\del\models\research\object_detection&gt;</pre> |
| <p>학습할 대상 물체 LabelMap 구현</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | <p>구성된 레이블을 Tensorflow Record 파일 형식으로 변환</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

도면6

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre> schedule {   step: 1200000   learning_rate: .000002 } momentum_optimizer_value: 0.9 use_moving_average: false gradient_clipping_by_norm: 10.0 fine_tune_checkpoint: "C:/Users/dell/models/research/object_detection/faster_rcnn_inception_v2_coco_2018_01_28/model.ckpt" from_detection_checkpoint: true # Note: The below line limits the training process to 200K steps, which we # empirically found to be sufficient enough to train the pets dataset. This # effectively bypasses the learning rate schedule (the learning rate will # never decay). Remove the below line to train indefinitely. num_steps: 200000 data_augmentation_options {   random_horizontal_flip {   } } }  train_input_reader {   tf_record_input_reader {     input_path: "C:/Users/dell/models/research/object_detection/train.record"   }   label_map_path: "C:/Users/dell/models/research/object_detection/training/labelmap.pbtxt" }  eval_config {   num_examples: 20   # Note: The below line limits the evaluation process to 10 evaluations.   # Remove the below line to evaluate indefinitely.   max_evals: 10 }  eval_input_reader {   tf_record_input_reader {     input_path: "C:/Users/dell/models/research/object_detection/test.record"   }   label_map_path: "C:/Users/dell/models/research/object_detection/training/labelmap.pbtxt"   shuffle: false   num_readers: 1 } </pre> |
| <p><b>파이프 라인 구성</b></p> <p>[어떤 모델, 어떤 매개 변수가 학습에 사용될지 정의]</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |

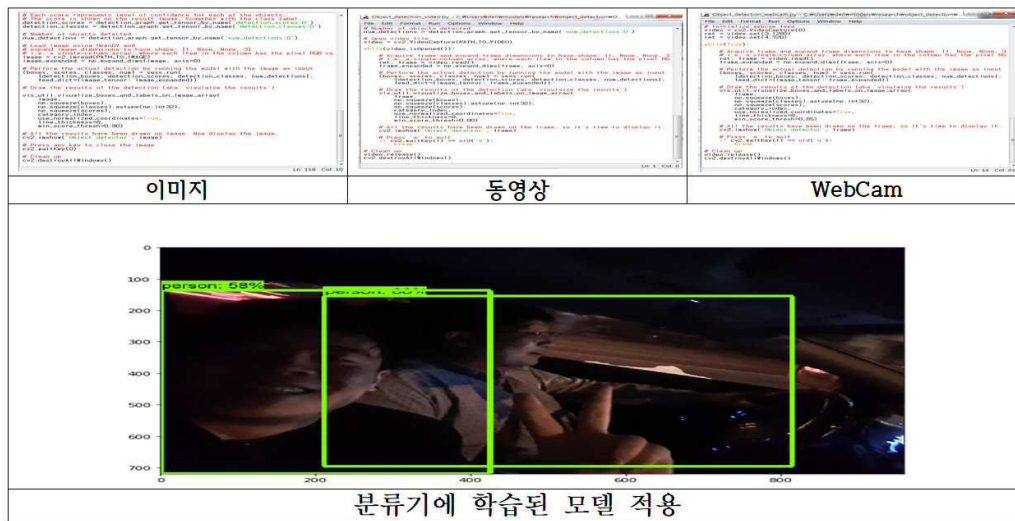
도면7

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                        |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|
| <pre> INFO:tensorflow:Starting Session. INFO:tensorflow:Saving checkpoint to path training/model.ckpt INFO:tensorflow:Starting Queues. INFO:tensorflow:Global Step/sec: 0 INFO:tensorflow:Recording summary at step 0. INFO:tensorflow:global step 1: loss = 2.6708 (5.383 sec/step) INFO:tensorflow:global step 2: loss = 3.0352 (0.251 sec/step) INFO:tensorflow:global step 3: loss = 3.4884 (0.204 sec/step) INFO:tensorflow:global step 4: loss = 2.9733 (0.193 sec/step) INFO:tensorflow:global step 5: loss = 2.2184 (0.191 sec/step) INFO:tensorflow:global step 6: loss = 2.0321 (0.554 sec/step) INFO:tensorflow:global step 7: loss = 2.0424 (0.211 sec/step) INFO:tensorflow:global step 8: loss = 2.0252 (0.208 sec/step) INFO:tensorflow:global step 9: loss = 2.0053 (0.184 sec/step) INFO:tensorflow:global step 10: loss = 1.9522 (0.193 sec/step) INFO:tensorflow:global step 11: loss = 1.8827 (0.197 sec/step) INFO:tensorflow:global step 12: loss = 1.2485 (0.196 sec/step) INFO:tensorflow:global step 13: loss = 1.0712 (0.193 sec/step) INFO:tensorflow:global step 14: loss = 1.6604 (0.189 sec/step) INFO:tensorflow:global step 15: loss = 1.2657 (0.192 sec/step) INFO:tensorflow:global step 16: loss = 1.4351 (0.193 sec/step) INFO:tensorflow:global step 17: loss = 1.2152 (0.192 sec/step) INFO:tensorflow:global step 18: loss = 1.1165 (0.197 sec/step) INFO:tensorflow:global step 19: loss = 1.6557 (0.192 sec/step) INFO:tensorflow:global step 20: loss = 1.7772 (0.200 sec/step) </pre> |                        |
| 학습 진행 현황                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | 손실그래프를 통해 데이터 총 손실을 확인 |

도면8

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <pre> Administrator: C:\WINDOWS\system32\cmd.exe - python export_inference_graph.py --input_type image_tensor --pipeline_config_path training/faster_rcnn_inception_v2_pets.config --trained_checkpoint_prefix training/... (tensorflow) C:\tensorflow\models\research\object_detection&gt; (tensorflow) C:\tensorflow\models\research\object_detection&gt;python export_inference_graph.py --input_type image_tensor --pipeline_config_path training/faster_rcnn_inception_v2_pets.config --trained_checkpoint_prefix training/model.ckpt-53698 --output_directory inference_graph WARNING:tensorflow:From C:\tensorflow\models\research\object_detection\core\box_predictor.py:380: calling reduce_mean (from tensorflow.python.ops.math_ops) with keep_dims is deprecated and will be removed in a future version. Instructions for updating: keep_dims is deprecated, use keepdims instead WARNING:tensorflow:From C:\tensorflow\models\research\object_detection\exporter.py:357: get_or_create_global_step (from tensorflow.contrib.framework.python.ops.variables) is deprecated and will be removed in a future version. Instructions for updating: Please switch to tf.train.get_or_create_global_step 2018-02-16 20:42:14.334552: I C:\tf_jenkins\workspace\rel-win\M\windows-gpu\PY\36\tensorflow\core\platform\cpu_feature_guard.cc:137] Your CPU supports instructions that this TensorFlow binary was not compiled to use: AVX AVX2 2018-02-16 20:42:14.542482: I C:\tf_jenkins\workspace\rel-win\M\windows-gpu\PY\36\tensorflow\core\common_runtime\gpu\gpu_device.cc:1105] Found device 0 with properties: name: GeForce GTX 1060 6GB major: 6 minor: 1 memoryClockRate(GHz): 1.7845 pciBusID: 0000:01:00.0 totalMemory: 6.00GiB freeMemory: 4.96GiB 2018-02-16 20:42:14.548125: I C:\tf_jenkins\workspace\rel-win\M\windows-gpu\PY\36\tensorflow\core\common_runtime\gpu\gpu_device.cc:1195] Creating TensorFlow device (/device:GPU:0) -&gt; (device: 0, name: GeForce GTX 1060 6GB, pci bus id: 0000:01:00.0, compute capability: 6.1) 2018-02-16 20:42:28.698368: I C:\tf_jenkins\workspace\rel-win\M\windows-gpu\PY\36\tensorflow\core\common_runtime\gpu\gpu_device.cc:1195] Creating TensorFlow device (/device:GPU:0) -&gt; (device: 0, name: GeForce GTX 1060 6GB, pci bus id: 0000:01:00.0, compute capability: 6.1) Converted 356 variables to const ops. 2018-02-16 20:42:35.660648: I C:\tf_jenkins\workspace\rel-win\M\windows-gpu\PY\36\tensorflow\core\common_runtime\gpu\gpu_device.cc:1195] Creating TensorFlow device (/device:GPU:0) -&gt; (device: 0, name: GeForce GTX 1060 6GB, pci bus id: 0000:01:00.0, compute capability: 6.1) </pre> |
| 학습이 완료되었을 마지막 단계에 고정 유추 그래프를 생성                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |

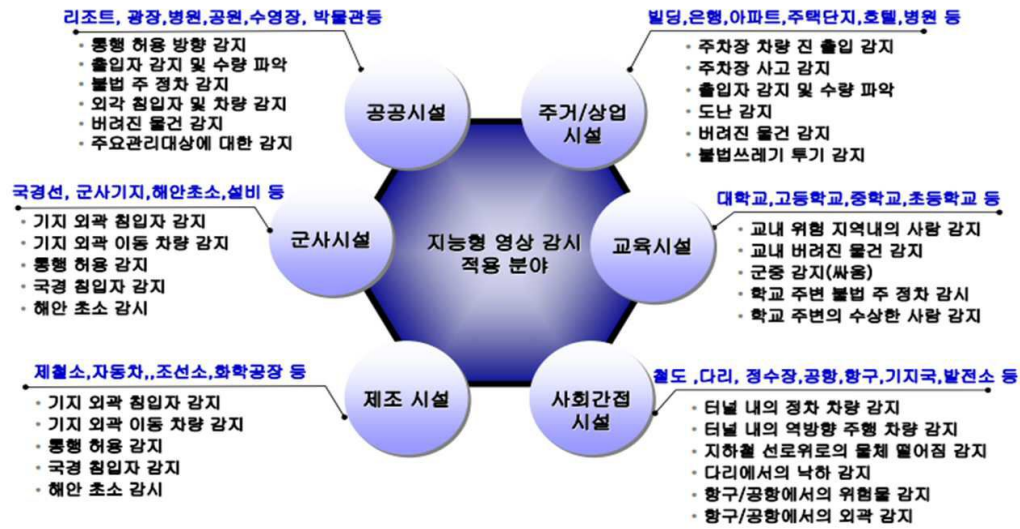
도면9



도면10



도면11



도면12

