

딥러닝 작업을 위한 오픈소스 기반 이미지 어노테이션 시스템 구축 및 공유

Implementation of Open Source Image Annotation System for Deep Learning Task

요 약

최근 4차 산업시대에 접어들면서 대용량 데이터 처리와 패턴 인식을 위한 기술이 발전하고 있다. 특히 대량의 이미지에서 여러 가지 특징을 추출하고 이를 딥러닝 작업에 활용하는 기술을 많이 사용하고 있다. 하지만 기존의 이미지 특성을 추출하는 도구들은 사용자가 직접 임의의 크기를 결정하고, 태그 이름을 입력해야하는 불편함을 가지고 있으며, 대량의 데이터를 다루고 처리해야하는 시스템 구현과 개발 기간의 부담이 있다. 본 논문에서는 서버 구현과 운영, 관리의 부담을 줄이는 서버리스 시스템을 구축하고, 오픈소스를 활용하여 학습 이미지를 편리하게 추출할 수 있는 시스템 개발과 플랫폼 공유를 제안한다.

1. 서 론

최근 대용량 데이터 처리와 패턴 인식을 위한 기술들이 발전해오고 있다. 특별히 이미지 분류(Image Classification)를 위하여 딥러닝(Deep Learning)이 활용되고 있다 [1]. 딥러닝은 이미지 분류를 위하여 대량의 학습 이미지들을 필요로 한다. 하지만 학습을 위하여 전체 이미지가 아닌 특성을 추출한 이미지 (64 X 64 pixel image, etc.)를 사용하는 경우가 많다. 이미지의 특성을 추출하기 위하여 LabelMe[2], Image Annotation Programme(IAP)[3], 그리고 Annotorious[4]와 같은 웹 기반의 오픈 소스 도구들이 공개되었다. LabelMe와 IAP 도구는 모듈 의존성 및 결합도(coupling)가 높아 모듈의 수정이 어렵다. 반면에 Annotorious 도구는 모든 기능이 모듈화 되어있어 새로운 모듈을 추가함으로써 구현 확장이 용이하다.

본 논문에서 제안하는 시스템은 딥러닝 작업을 위한 대량의 학습 이미지들에 대하여 일정한 크기의 특성 추출과 어노테이션 입력의 편리성에 있다. 하지만 기존의 도구들은 사용자가 특성 추출을 위하여 임의의 사각형 크기를 직접 드래그하여 그 크기를 결정해야 한다. 그러므로 추출된 각각의 이미지들의 크기가 서로 다를 수 있다. 또한, 태그 이름을 매번 입력해야하는 불편함도 가지고 있다. 따라서 본 논문에서는 Annotorious 도구를 확장한 딥러닝 작업을 위한 이미지 어노테이션 시스템을 제안한다.

2. 오픈소스를 활용한 이미지 어노테이션 구현

Annotorious는 전체 소스를 수정할 필요 없이 구현이 필요한 기능만 모듈로 만들어 오픈소스 프로젝트로 쉽게 기여할 수 있다.

도구는 <https://github.com/annotorious/annotorious/releases/tag/v0.6.4> 에서 다운로드 받을 수 있다.

```
<div id="block">

</div>
```

(그림 1) Image Element

Annotorious는 웹 페이지(HTML) 구성 요소 중 이미지를 보여주기 위한 요소 ''의 태그 ID를 makeAnnotatable 메소드가 참조한다.

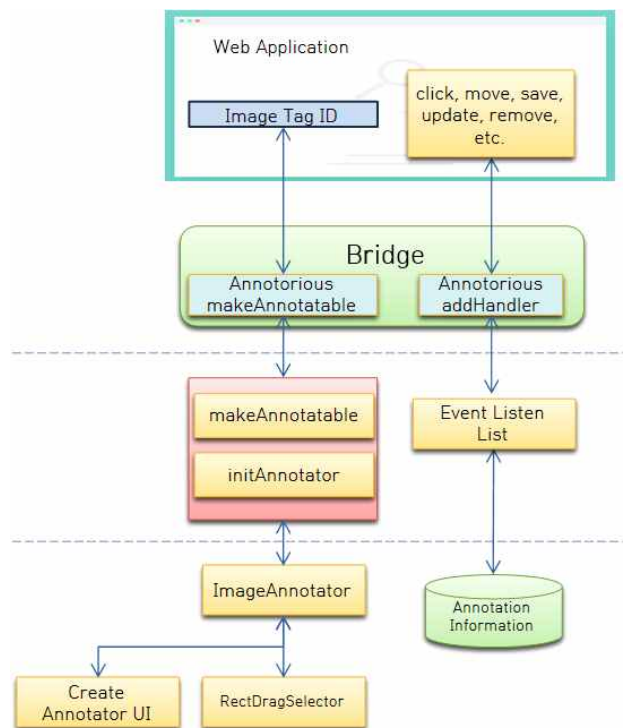
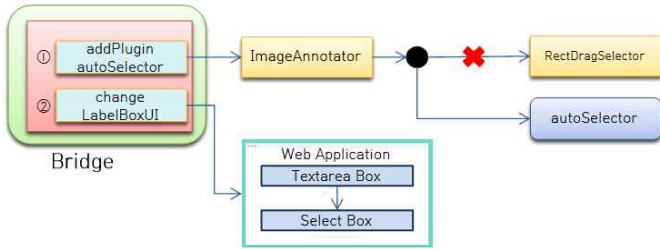


그림 2 Annotorious 순서도

참조한 이미지 태그 ID로 ImageAnnotator 메소드가 접근하여 (그림 2)와 같은 순서도를 따라 어노테이션 환경을 구축한 뒤, addHandler 메소드를 사용하여 어노테이션 정보를 저장하거나 수정하는 기능을 이곳에 구현한다.

2.1 플러그인 모듈 구현



(그림 3) 모듈 구현과 적용 순서도

본 논문에서 제안하는 시스템은 첫 번째로 마우스 클릭 지점을 중심으로 사각형(e.g, 64 x 64, etc)의 이미지 특징을 추출하는 플러그인 모듈 구현이며, 두 번째는 추출하려는 특징을 표현할 수 있는 태그(Tag) 이름을 미리 만들어 둔 List에서 선택할 수 있도록 구현하는 것이다.

```

annotorious.plugin.autoSelector.Selector.prototype.drawRect =
function (click_x, click_y) {
  if (this._drawLocking == false) {
    this._g2d.strokeStyle = "#35E5F1"; // 청색
    this._g2d.lineWidth = 2;

    var x = click_x - 32;
    var y = click_y - 32;
    var w = 64;
    var h = 64;

    if (x < 0) {
      x = 0;
    }
    if (y < 0) {
      y = 0;
    }
    this._g2d.strokeRect(x, y, w, h);
  }
}

```

(그림 4) 64 x 64 정사각형 소스코드

addPlugin으로 모듈을 추가할 경우 (그림 3)의 ①처럼 기존의 사용하고 있던 모듈을 본 논문에서 구현한 모듈 autoSelector로 대체한다.

(그림 4)는 추출하려는 이미지의 특징을 클릭 할 때, 자동으로 64 x 64 크기의 정사각형 테두리를 그려주는 autoSelector 모듈의 기능을 나타낸다.

```

function annotorious_changeToTagList() {

  $('annotorious-editor-text.goog-textarea')
  .replaceWith("<select id='selectTagName'>" + $(this).text() + "</select>");

  var tagNames = ["frill", "red", "goldEmbroidery", "check", "line", "black"];

  tagNames.forEach(tag => {
    var typeOption = document.createElement('option');
    var tagOption = document.createTextNode(tag);
    typeOption.appendChild(tagOption);

    document.getElementById("selectTagName").appendChild(typeOption);
  });
}

```

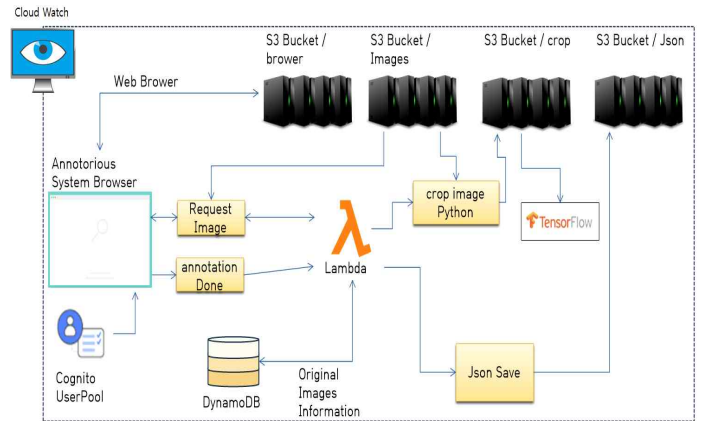
(그림 5) 태그 입력 UI 변경

②는 추출하려는 특징의 태그(Tag) 이름을 직접 입력 하는 방식에서 미리 만들어 둔 태그 List에서 선택할 수 있도록 UI를 변경한다.

3. 딥러닝 작업을 위한 Serverless 시스템 구축

본 논문에서는 서버리스 시스템 구축을 제안한다. 서버리스 시스템은 별도의 서버를 구매하거나, 설치, 그리고 관리할 필요 없이 AWS에서 지원하는 다양한 시스템 어플리케이션을 API Gateway와 람다(Lambda)를 활용하여 연동한다.

서버리스 시스템 구축으로 서버 관리와 운영의 대한 부담을 줄일 수 있으며, 딥러닝 작업과 이미지 추출 구현과 같은 어플리케이션 개발에 집중할 수 있어 더욱 효과적으로 시스템을 구축할 수 있다.



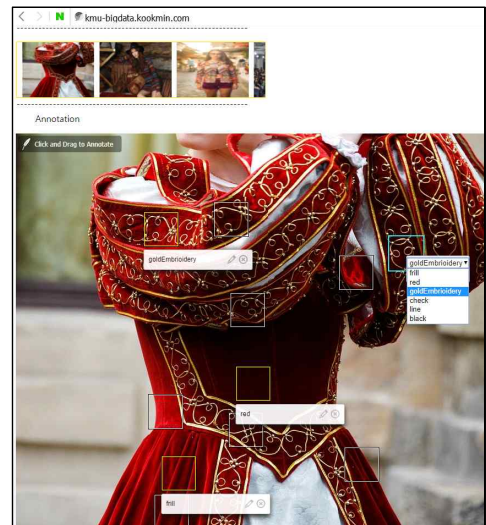
(그림 6) 서버리스 시스템 구축

S3(Simple Storage Service) 버킷은 웹 페이지 폼과 스크립트 소스를 저장하고 있는 브라우저(Browser) 버킷과 원본 이미지 정보들을 저장하고 있는 이미지(Images) 버킷 및 추출한 이미지가 저장되는 Crop 버킷, 그리고 어노테이션 결과를 저장하는 Json 버킷이 있다.

어노테이션 시스템에 접속하는 사용자들은 브라우저 버킷으로 접근하며, 각 작업은 람다(Lambda)를 사용하여 서로 다른 어플리케이션을 연동한다.

람다는 주기적으로 이미지 버킷에 원본 이미지들과 Json 버킷에 저장된 어노테이션 정보를 활용하여 Python 코드로 작성된 Crop 프로그램으로 이미지 특징을 추출 후 Crop 버킷에 저장한다.

4. 구현 결과



(그림 7) 구현 결과

5. 결론 및 향후 연구

참 고 문 헌

- [1]:김지원, 표현아, 하정우, 이찬규, 김정희 (2015). 다양한 딥러닝 알고리즘과 활용. 정보과학회지, 33(8), 25-31.
- [2]:B. C. Russell, A. Torralba, K. P. Murphy, W. T. Freeman. LabelMe: a Database and Web-based Tool for Image Annotation. International Journal of Computer Vision, 77(1-3):157-173, 2008
- [3]:https://github.com/frederictost/images_annotation_programme
- [4]:<https://github.com/annotorious/annotorious>