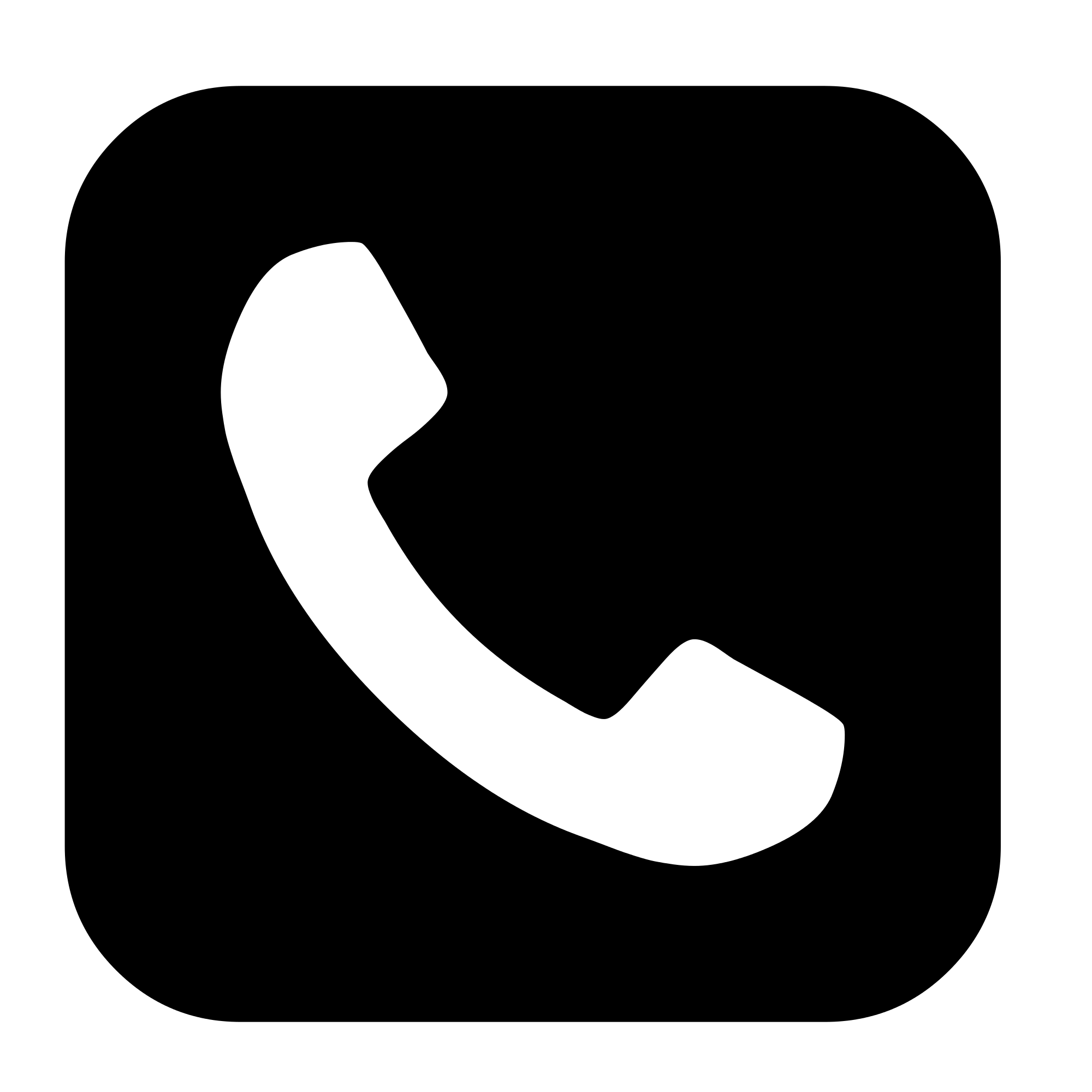
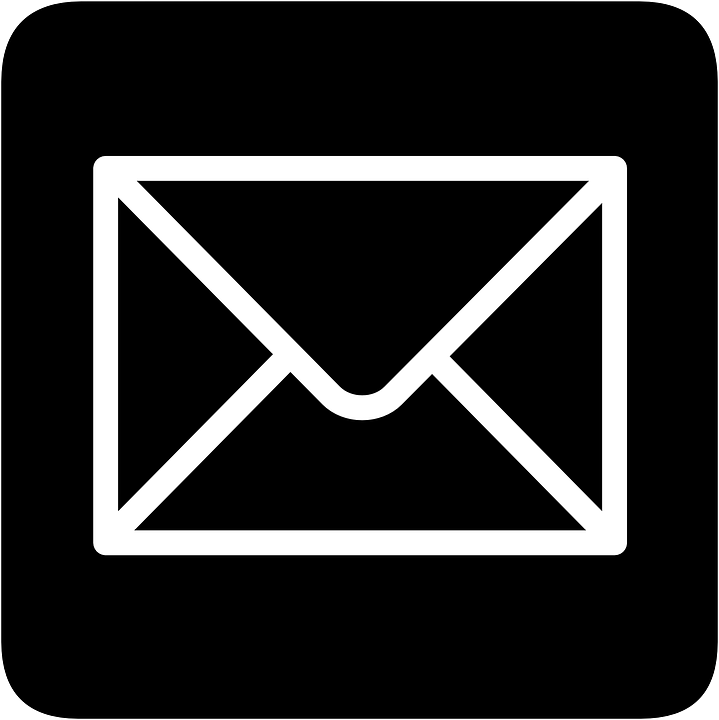
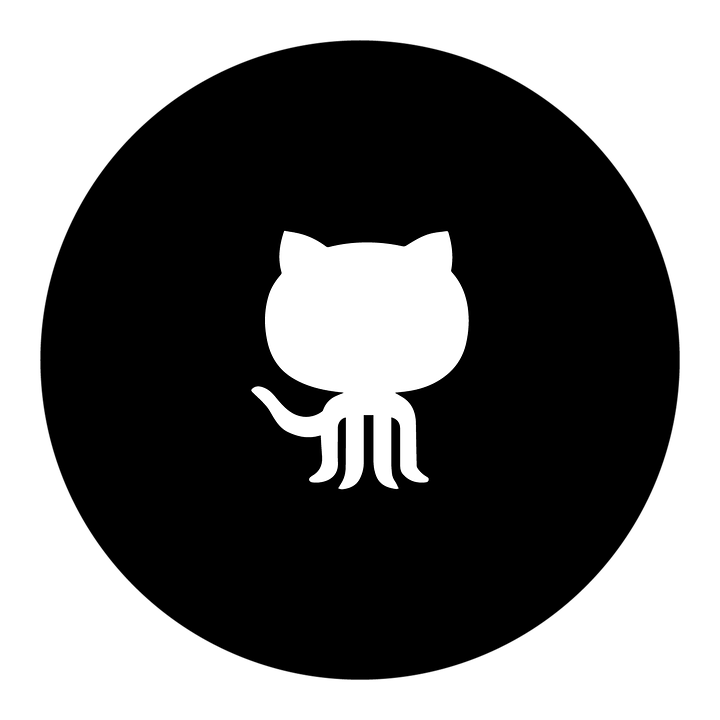
민지호 

 010-2170-1871

 [21300271@handong.edu](mailto:21300271@handong.edu)

 <https://github.com/JihoMin>

|  |  |
| --- | --- |
| 학력 | 배재고등학교, 서울2010년 3월 - 2013년 2월한동대학교, 포항2013년 3월 - 2019년 2월(졸업 예정) |
| 프로젝트 | iShopping -Mobilenet을 이용한 오프라인 대형 매장 의류 쇼핑 솔루션 2018년 1월 - 2018년 6월   * Ionic framework, Node.js, AWS EC2, MongoDB * Mobilenet 모델을 이용한 비슷한 옷 분류, 추천  다중 오믹스-임상정보 통합을 통한 정밀의학 플랫폼 구축 2018년 6월 - 현재   * Vue.js, Node.js, AWS EC2, MySQL * 데이터 통계분석을 위한 데이터베이스 구축 * 데이터 입력, 조회를 위한 웹사이트 개발 |
| 활동 | 시선 - 소외된 이웃을 위한 IT 솔루션 연구 학회2017년 4월 - 2018년 6월  * 시각장애인을 위한 쇼핑몰 어플리케이션 * 부학회장(2018-1학기)  Bio Data Lab2018년 6월 - 현재  * 건강검진 데이터 베이스 구축 * 웹사이트 개발 |
|  |  |
| 자기소개서 | **즐거운 몰입의 경험**  머신러닝 수업에서 python으로 KNN classifier를 구현한 적이 있습니다. 직접 구현한 코드의 분류 결과와 scikit-learn의 KNN 라이브러리의 결과를 비교해 보니 테스트 정확도는 같게 나오지만, 속도가 훨씬 느리다는 문제가 있었습니다. 실행속도의 차이를 만드는 요소가 어떤 것인지 궁금해졌고 scikit-learn의 KNN 코드를 살펴보게 되었습니다. 코드를 분석하면서 euclidean distance matrix를 구하는 연산 과정에서 제가 구현한 코드와 두 가지 차이점을 발견할 수 있었습니다. 첫 번째는 루트 연산에서 발생하는 시간을 줄이기 위해 squared euclidean distance를 사용한다는 점이었고, 두 번째는 행렬의 곱셈 부분을 c로 구현하여 속도를 높였다는 점이었습니다. 저는 squared euclidean distance를 사용하고 Cython을 이용해 코드를 c언어로 컴파일하는 방법을 적용하여 실행 시간을 세 배가량 단축할 수 있었습니다.  단순하고 사소한 발전이었지만 머신러닝 알고리즘을 구현할 때 성능 개선을 위해 고려해야 할 요소들에 대해 고민해볼 수 있는 시간이었습니다. 또한, 이 경험을 통해 제가 어떤 문제에 호기심을 느끼며 몰입할 수 있는지 알 수 있게 되었습니다.  **iShopping**  iShopping은 사용자가 찍은 의류 사진과 매장 데이터베이스에 있는 의류 사진을 매칭하여 쇼핑 정보를 손쉽게 저장하고, 비슷한 옷을 추천받을 수 있도록 기획한 모바일 애플리케이션 프로젝트입니다. 의류 사진을 카테고리별로 분류하고, Mobile net을 이용해 학습하여 의류 매칭을 구현하였습니다. 처음에는 open data를 이용하여 분류하였는데, 데이터가 많은 대신 카테고리별 의류 사진의 일관성이 부족하여 테스트 정확도가 매우 낮게 나왔습니다. 이를 개선하기 위해 python의 Beautiful Soup을 이용하여 자체적으로 나눈 카테고리별로 이미지를 수집했고, 정확도를 크게 개선할 수 있었습니다. 비록 모델을 가져다 사용하는 수준에 그쳤지만, 딥러닝을 이용해 스스로 정한 문제를 해결해 볼 수 있는 경험이었습니다. |