ResNet50 전이학습을 활용한 손동작 인식 기반 가위바위보 게임 구현

박창준*, 김창기O, 손성규*, 이경진*, 유희경*, 곽정환(교신저자)*
*한국교통대학교 AI로봇공학과,
O한국교통대학교 AI로봇공학과
e-mail: {1726018, 1715086, 1915153, 1926034, jgwak}@ut.ac.kr*, 1715071@ut.ac.krO

Implementation of hand motion recognition-based rock-paper-scissors game using ResNet50 transfer learning

Changjoon Park*, Changki KimO, Seongkyu Son*, Kyoungjin Lee*,
Heekyung Yoo*, Jeonghwan Gwak(Corresponding Author)*
*Dept. of AI Robotics, Korea National University of Transportation,
ODept. of AI Robotics, Korea National University of Transportation

● 요 약 ●

GUI(Graphical User Interface)를 대신하는 차세대 인터페이스로서 NUI(Natural User Interace)에 기대가 모이는 것은 자연스러운 흐름이다. 본 연구는 NUI의 손가락 관절을 포함한 손동작 전체를 인식시키기 위해 웹캠과 카메라를 활용하여 다양한 배경과 각도의 손동작 데이터를 수집한다. 수집된 데이터는 전처리를 거쳐 데이터셋을 구축하며, ResNet50 모델을 활용하여 전이학습한 합성곱 신경망(Convolutional Neural Network) 알고리즘 분류기를 설계한다. 구축한 데이터셋을 입력시켜 분류학습 및 예측을 진행하며, 실시간 영상에서 인식되는 손동작을 설계한 모델에 입력시켜 나온 결과를 통해 가위바위보 게임을 구현한다.

키워드: 합성곱 신경망(CNN), ResNet50, 전이학습, 손동작 인식, 가위바위보 게임

I. Introduction

최근 PC와 VR, AR 등과 융합된 NUI(Natural User Interface)[1]는 눈동자 인식, 손동작 인식 등 동작 인식이 가능하다. 인간 손에는 손가락 관절이 존재하므로 손동작은 눈동자로 표현할 수 있는 동작보다 수가 많으며 효율적이다. 따라서 손동작 인식은 마우스나 키보드 같은 부가적인 장치 없이 사용자와 컴퓨터 간의 상호작용을 원활하게 할 수 있는 효과적인 방법이다.

본 논문에서는 손가락 관절을 포함한 손 전체 동작을 인식시키기 위하여 시각적 이미지를 분석하는 데 쓰이는 인공 신경망 중 하나인 합성곱 신경망(Convolutional Neural Network)[2] 알고리즘을 사용한다. 이는 다차원 배열 데이터 처리가 가능하며 일반적인 신경망과는 다르게 이미지에서 특징을 추출해 처리하므로 컬러 이미지 처리에 특화되어 있다. 또한 부족한 데이터셋을 보완하기 위해 전이학습을 진행하고 방대한 이미지를 통해 학습한 ResNet50 모델[3]을 활용하여 합성곱 신경망을 설계한다. 이러한 합성곱 신경망은 다음과 같은 과정으로 진행된다.

첫째, 웹캠 및 카메라를 통해 여러 각도와 배경으로 영상을 수집하고, 프레임 간격으로 이미지화하여 라벨을 붙여줌으로써 데이터셋을 구축한다. 둘째, 구축된 데이터셋은 ResNet50 모델을 사용하여 전이학습한 합성곱 신경망을 활용하여 분류학습을 진행한다. 셋째, 웹캠을 통해 실시간으로 입력받은 손동작 이미지를 분류된 클래스에 맞게 예측이 진행되는지를 평가한다. 마지막으로 설계된 모델에 손동작을 인식하여 나온 예측 결과들을 통해 가위바위보 게임[4, 5]을 구현한다.

II. Preliminaries

2.1 CNN (Convolutional Neural Network)

CNN[3]은 Deep Learning 알고리즘 중 하나로 이미지 처리에 뛰어난 성능을 보이는 신경망이다. CNN의 기본 구조는 Convolution Layer와 Pooling Layer를 반복적으로 적용하여 여러 겹 쌓는 특징 추출 부분, Fully Connected Layer를 구성하고 출력층에 Softmax를

적용한 분류 부분으로 구성된다.

학습 과정은 다음과 같이 진행하였다. 먼저 필터를 입력 이미지에 교차 적용하여 곱 연산을 하고 활성화 함수를 거친 후 결과 값을 추출한다. 결과 값은 1차원 배열 데이터로 변환되어 Fully Connected Layer를 거쳐 분류된다. 본 논문에서는 Input_image로 여러 배경에서 촬영한 가위바위보 영상을 224*224 이미지로 데이터 전처리하여 사용하며, 흔들림이 있는 Noise Data는 제거한다. 또한 총 4개의 분류를 얻기 위해 Softmax 출력값을 조정한다.

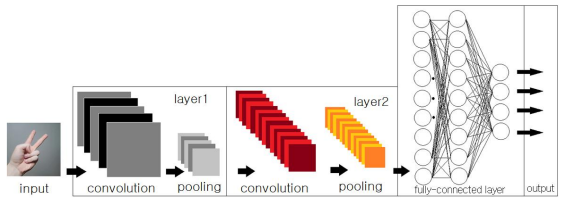

Fig. 1. Convolution Neural Network의 기본적인 구조

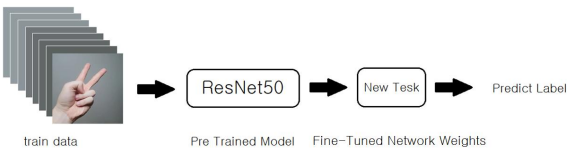

Fig. 2. ResNet50기반 전이학습

2.2 ResNet50기반 전이학습

관련 작업에 대해 이미 훈련된 모델의 일부를 가져와 새 모델에서 재사용함으로써 손쉽게 데이터를 구축하는 기술이다. 주로 학습데이터가 부족한 분야의 모델 구축을 위해 사용한다. ResNet50은 Image Net 데이터베이스의 백 만개가 넘는 영상에 대해 훈련한 50개의 층을 갖는 CNN 모델이다. 본 논문에서는 ResNet50의 가중치를 불러와 층을 추가하고 가위바위보 분류기를 학습한다. 이번 연구에서 구축한 입력 데이터의 수가 가위바위보 행동별 3천 장으로 학습을 원활히 진행하기 힘들다고 판단하였다. 따라서 ResNet50을 활용한 전이학습을 통해서 입력 데이터 부족으로 인한 모델 성능 문제를 해소한다.

III. The Proposed Scheme

본 연구에서는 가위바위보 손동작의 분류를 위해 데이터를 수집 및 전처리하는 과정을 거친다. 이때 생성된 데이터셋을 합성곱 신경망에 입력하여 손동작을 분류할 수 있도록 학습시키고 실제 데이터를 분류한다. 또한 가위바위보 알고리즘을 구성하고 게임을 직관적으로 진행할 수 있도록 python의 tkinter 모듈을 이용하여 GUI를 구성한다. 연구는 1080p camera, CPU-i9 10900k, GPU-RTX3090, RAM-16GB, cuda11.4, cudnn11.4, python3.9.5와 python의 tkinter, tensorflow-gpu2.5, OpenCV4.5.4 환경에서 진행한다.

1. 데이터수집 및 전처리

가위바위보에 대한 손동작을 합성곱 신경망 모델에 학습시키기 위한 데이터 수집 및 전처리하는 단계이며 데이터의 수집은 카메라(스마트폰 및 웹캠)를 통해 진행된다.

가위바위보의 데이터셋 종류는 가위 1(엄지와 검지), 가위 2(검지와 중지), 바위, 보 총 4개로 구성된다. 해당 4개의 손동작을 카메라 앵글 중앙에 각각 비추어 영상을 촬영 및 저장하고, 위 영상 데이터를 OpenCV를 이용하여 10프레임당 1장의 이미지 데이터로 변환 및 데이터셋화 한다. 해당 데이터셋을 합성곱 신경망 모델에 입력하기 위해 영상의 중앙에서부터 정사각형 모양으로 이미지를 자르고 벡터의 차원을 224*224*3의 크기로 변환하여 저장한다.

남은 전처리는 ResNet50 전이학습 모델에서 지원하는 preprocess_input 함수를 이용한다. 가위 1, 가위 2, 바위, 보의 라벨당 데이터 개수는 (표 1)과 같이 각 라벨마다 3천 개로 총 1만 2천 개의 데이터로 구성되며, 전체 데이터의 80%는 특징 데이터, 나머지 20%는 검증 데이터로 나누어 모델 학습에 사용한다. 손동작 영상은 합성곱 신경망 모델의 편향 학습을 막기 위해 다양한 손의 각도와 배경에서 (그림 3)과 같이 촬영한다.

Table 1.

	가위1	가위2	바위	보	전체
특징	2,400	2,400	2,400	2,400	9,600
검증	600	600	600	600	2,400
전체	3,000	3,000	3,000	3,000	12,000

Fig. 3.

2. 합성곱 신경망(CNN) 모델 제작 및 학습

현 단계에서는 앞서 구축한 손동작 이미지 데이터셋을 분류하기 위해 합성곱 신경망 모델을 생성하고 모델 학습을 진행한다. 해당 모델은 ResNet50의 예측률이 뛰어난 가중치를 이용해 합성곱 신경망 분류기를 전이 학습시켜 예측률을 높인다.

먼저 tensor flow에서 제공하는 ResNet50 모델을 불러와 weights='imagenet' 및 include_top=false 옵션을 적용하여 ResNet50의 가중치인 imagenet을 학습에 사용하고 말단의 Fully-connected layer를 제거한다. 이후 앞서 정의한 손동작 데이터셋을 구별할 수 있도록 4개의 class를 출력하는 합성곱 신경망 분류기를 만든다. 분류기는 Flatten layer/Fully-con nected layer (activation='relu)/output layer (activaion= 'softmax') 순서로 구성된다. 합성곱 신경망 분류기는 (그림 4)와 같이 Fully-connected

layer를 제거한 ResNet50 모델의 말단에 연결된 후, 입력받은 7x7x 2048의 차원의 벡터를 가위 1, 가위 2, 바위, 보 손동작의 4개의 라벨로 분류하여 출력한다. 구성을 완료한 합성곱 신경망 모델은 손실함수로 catego rical_crossentropy, 최적화 함수로 Adam을 사용하여 컴파일한다. 위 과정을 모두 진행한 후 합성곱 신경망 분류기를 ResNet50의 가중치인 imagenet과 가위1, 가위2, 바위, 보의 손동작 데이터(224x224x3의 차원을 가지는 이미지)를 입력받아 학습을 진행한다. (그림 5)을 통해 전이학습이 완료된 모델의 검증데이터 정확률이 96으로 좋은 성능을 보이는 모습을 확인할 수 있다.

```
In [10]: cnn.summary()
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
resnet50 (Functional)        (None, 7, 7, 2048)        23587712
_________________________________________________________________
flatten (Flatten)            (None, 100352)            0
_________________________________________________________________
dense (Dense)                (None, 1024)              102761472
_________________________________________________________________
dense_1 (Dense)              (None, 4)                 4100
=================================================================
Total params: 126,353,284
Trainable params: 126,300,164
Non-trainable params: 53,120
_________________________________________________________________
```

Fig. 4. ResNet50을 이용한 전이학습 CNN 모델

```
Epoch 9/10
2400/2400 [==============================]
val_loss: 0.3986 - val_accuracy: 0.9504
Epoch 10/10
2400/2400 [==============================]
val_loss: 0.2947 - val_accuracy: 0.9604
정확률은 96.04166746139526
```

Fig. 5. 전이학습 완료된 CNN 모델의 정확률

3. 학습 완료된 분류기의 실제 데이터 검증 결과

전이학습이 완료된 합성곱 신경망 분류기를 이용하여 실전 데이터를 검증한다. (그림 6)는 가위바위보 분류 모델 학습에 사용되지 않은 테스트 데이터셋의 모습이다. 왼쪽 위에서부터 가위 1, 가위 2, 바위, 보의 손동작 이미지가 각자 다른 배경과 손의 각도를 가지고 있으며, 하나의 라벨당 50개의 이미지 데이터로 이루어져 있다.

Fig. 6. 성능 검증에 사용된 test datasets

```
2/2 [==============================] - 1s 61ms/step - loss: 1.3051 -
accuracy: 0.9600

정답 label
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
예측 label
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

가위1(0) label의 50개의 data중 총 2개 오답.
오답 [index,label]: [[18, 1], [30, 3]]
정답률은 0.9599999785423279  loss는 1.3050732612609863
```

Fig. 7. 가위1에 대한 실제 데이터 검증

```
2/2 [==============================] - 1s 61ms/step - loss: 0.3693 -
accuracy: 0.9600

정답 label
[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
예측 label
[1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]

가위2(1) label의 50개의 data중 총 2개 오답.
오답 [index,label]: [[8, 0], [37, 0]]
정답률은 0.9599999785423279  loss는 0.3693004548549652
```

Fig. 8. 가위2에 대한 실제 데이터 검증

```
2/2 [==============================] - 1s 61ms/step - loss: 4.7684e-08 -
accuracy: 1.0000

정답 label
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]
예측 label
[2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2]

바위(2) label의 50개의 data중 총 0개 오답.
오답 [index,label]: []
정답률은 1.0  loss는 4.768367389829109e-08
```

Fig. 9. 바위에 대한 실제 데이터 검증

```
2/2 [==============================] - 1s 61ms/step - loss: 3.3379e-08 -
accuracy: 1.0000

정답 label
[3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3]
예측 label
[3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3]

보(3) label의 50개의 data중 총 0개 오답.
오답 [index,label]: []
정답률은 1.0  loss는 3.3378579900045224e-08
```

Fig. 10. 보에 대한 실제 데이터 검증

(그림 7, 8, 9, 10)은 순서대로 가위 1, 가위 2, 바위, 보에 대한 각각 50개의 손동작 테스트 데이터를 검증한 모습이다.

위 결과를 통해 가위바위보 분류 모델의 테스트 데이터셋 검증 성능을 확인해 볼 수 있다. 가위 1과 가위 2는 50개의 테스트 데이터 중 48개의 예측 성공으로 96의 정확률을 보였으며, 바위와 보는 현재의 테스트 데이터를 모두 맞추는 모습을 보였다.

4. 가위바위보 게임 개발

카메라를 이용하여 가위바위보 손동작 정보가 담긴 실시간 영상 스트리밍 데이터를 특징 데이터로 받아와 전처리한다. 이후 전이학습이 완료된 합성곱 신경망 분류기로 예측한 뒤 해당 값을 NUI로 사용하는 가위바위보 게임을 개발한다. 가위바위보의 승패를 알 수

있는 알고리즘을 간단하게 구현하였으며, 상대방의 가위바위보 라벨 값은 랜덤하게 적용된다.

자신이 카메라로 촬영하고 있는 손동작 데이터를 직관성 있게 확인할 수 있도록 OpenCV를 이용하여 카메라에 비추고 있는 손의 모습이 보일 수 있게 영상 처리를 하고, 합성곱 신경망 분류기가 예측한 라벨을 영상 위에 출력한다. 또한 스트리밍 데이터를 예측할 때 손동작을 잘못 인식할 수 있으므로 게임 시작 버튼이 작동한 후 손동작 데이터를 3초간 10개의 이미지를 처리 및 예측한다. 이후 예측한 확률값을 모두 더하여 확률이 제일 높은 라벨을 최종 예측 라벨로 선정한다. 선정된 라벨값과 랜덤하게 정해진 상대방 라벨 정보를 가위바위보 승패 알고리즘에 입력한 후 출력값을 중앙에 띄운다. (그림 11)은 가위바위보 게임을 실행한 초기 화면이다.

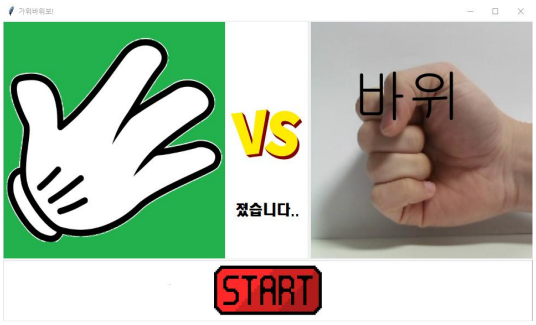

Fig. 11. 가위바위보 게임 실행 영상

IV. Conclusions

개발한 게임을 테스트하기 위해 배경과 각도를 달리하여 게임을 진행한다. 배경은 흰 배경과 모니 상에서 띄운 풍경 사진을 배경으로 변화를 주고 각도는 정면, 윗면, 아랫면, 손바닥, 손등, 대각선 총 6개의 각도로 구분하여 테스트한다. 웹캠에 가위바위보 행동을 인식하고 하단부에 start 버튼을 누르면 3초 뒤에 컴퓨터가 무작위로 낸 가위바위보와 비교하여 화면에 결과를 표시한다. 테스트를 계속한 결과, 가위바위보 중 각도에 구애받지 않는 바위의 인식률이 가장 높았고 가위는 손바닥 각도일 때 가장 인식률이 높았지만, 화면상 앞에 위치한 손가락에 의해 뒤의 손가락이 가려지는 아랫면, 윗면 각도에서는 가위와 보를 명확히 구분하지 못하는 결과를 확인했다. 보도 마찬가지로 정면, 아랫면, 윗면에서 인식률 저하를 확인했고 손가락 사이의 거리를 크게 할수록 인식률이 높아지는 것을 알 수 있다.

Fig. 12-1. 하얀 배경에서 연구한 결과-가위

Fig. 12-2. 하얀 배경에서 연구한 결과-바위

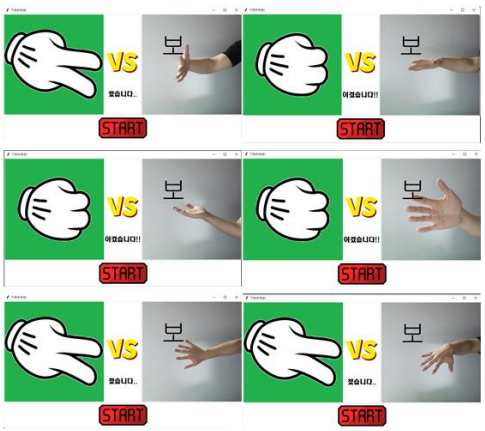

Fig. 12-3. 하얀 배경에서 연구한 결과-보

Fig. 13-1. 배경 변화를 주며 연구한 결과-가위

Fig. 13-2. 배경 변화를 주며 연구한 결과-바위

Fig. 13-3. 배경 변화를 주며 연구한 결과-보

더불어 모델 학습 부분에서 가위의 데이터셋을 총 2개(가위 1, 가위 2)로 구분했으나, (그림 14)를 통해 학습시키지 않은 중지와 약지, 약지와 소지 형태의 가위도 인식하는 모습을 확인할 수 있었다.

Fig. 14-1. 중지와 약지

Fig. 14-2. 약지와 소지

본 논문에서는 합성곱 신경망 모델을 이용하여 손동작을 인식하는 알고리즘 개발 및 NUI를 적용한 가위바위보 게임 시스템을 제안한다.

가위 1, 가위 2, 바위, 보를 각 3천 장, 총 1만 2천 장으로 구성된 데이터셋과 ResNet50의 가중치를 활용하여 합성곱 신경망 분류기의 학습을 진행한다.

해당 분류기 학습결과, 가위바위보를 분류하는 합성곱 신경망 모델은 약 96%의 정확률을 얻었다. 또한 바위의 인식률이 가장 높았고 가위의 인식률이 가장 낮음을 확인한다. 즉 가위와 보, 정면에서 바라보는 가위나 바위 등 손가락 관절이 겹치는 부분을 미세하게 인식하지 못하는 모습을 보인다. 해당 부분에 있어 인식률을 높이는 방향으로 학습모델을 개선하기 위해서는 기존보다 더 많은 양의 이미지 데이터가 필요하므로 더욱 다양한 각도와 배경에서의 이미지 데이터를 수집해야 한다. 이러한 방법을 통해 더 나은 모델로 개선하면 미세한 손동작도 인식할 수 있으므로 손동작으로 제어하는 부분에 있어 낮은 오류를 보여줄 것이고, 기존 터치를 중심으로 하는 GUI보다 더욱 큰 편리함을 가져와 준다.

또한 VR, AR 등에서도 손동작으로 여러 콘텐츠를 이용하는 부분에 있어서 높은 인식률로 부드럽고 세세한 손동작들을 수행할 수 있으므로 더 높은 질과 다양한 콘텐츠들을 구현해낼 것으로 기대된다.

ACKNOWLEDGEMENT

This results was supported by "Regional Innovation Strategy (RIS)" through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(MOE) (2021RIS-001).

REFERENCES

[1] Hee-Sun Choi, "A Study on NUI with Fingertip Detection and Tracking", Chung-Ang University Seoul Campus Academic Information Center, pp.12~24, 2012

[2] Hyun-soo Lee, “A Structure of Convolutional Neural Networks for Image Contents Search” Chung-Ang University Seoul Campus Academic Information Center, pp.1~14, 2018

[3] Sung-Wook Park, Do-Yeon Kim, "Comparison of Image Classification Performance in Convolutional Neural Network according to Transfer Learning", Journal of Korea Multimedia Society Vol. 21, No. 12, December 2018, pp.1388-1393, 2018

[4] Jeong-min Seo, Byeong-ju Kim, Hyeong-man Moon , Chang-sun Park, Jung-hwan Hwang, “Paper-scis sors-stone Game System Used by Hand Image Recognition Technique” Korea Multimedia Society, pp.510~512 ,2010

[5] Yeon-Su Jang, Da-Ye Kim, Dong-Jin Park, YunSung Han, Soobin Jeon, Dongmahn Seo, “A Rock-paper -scissors Game Using Hand Image Recognition Technology based on Artificial Neural Network“, Korea Information Processing Society, pp.659-662, 2020