

발레 자세 교정을 위한 발레 동작 코칭 시스템

손예진¹ · 최승호^{2*}¹송실대학교 기계공학부 학사과정^{2*}광운대학교 소프트웨어사업단 초빙교수

Ballet Posture Coaching System for Correction of Ballet Posture

Yea-Jin Shon¹ · Seung-Ho Choi^{2*}¹Bachelor Course, Department of Mechanical Engineering, Soongsil University, Seoul 06978, Korea^{2*}Visiting Professor, National Program for Excellence in Software, Kwangwoon University, Seoul 018967, Korea

[요약]

발레는 몸의 가동 범위가 넓은 운동인 만큼 정확하지 않은 자세로 운동을 하게 되면 부상을 입을 수 있고, 운동 효과를 느끼기 어렵다는 문제점이 있다. 이에 사용자가 올바른 자세로 운동을 할 수 있도록 돕는 발레 코칭 시스템을 제안한다. 제안한 방법은 15개의 관절이 레이블링 된 발레 동작을 기반으로 ResNet 기반의 Simple pose 모델과 Open pose의 BODY-25 모델을 사용하여 관절의 좌표를 검출한다. 검출한 관절 좌표의 분포를 히스토그램의 분위 수 분석을 통해 분석했다. 6가지 분류기를 통해 3가지 발레 동작 분류를 4가지 성능 지표를 통해 분석한다. Gradient Boosting Classifier가 비교한 모델 중 가장 최적의 성능 획득했다. 특징 중요도 분석 결과, 발목 관절이 중요하게 사용됨을 확인했다. 또한 4가지 특징 추출 방법에 따른 모델의 성능을 분석했다. 그리고 Odds ratio를 구하여 동작 별 예측 성공 비율을 비교했다. 제안한 방법이 발레 동작 코칭을 할 수 있음을 확인했다.

[Abstract]

Since ballet is an exercise with a wide range of movement, if you exercise with an incorrect posture, you may get injured and it is difficult to feel the effect of the exercise. We propose a ballet coaching system that helps users to exercise with correct posture. The proposed method detects joint coordinates using the ResNet-based Simple pose model and the Open pose BODY-25 model based on ballet motions labeled with 15 joints. Analysis of the quantiles of the histogram was performed on the distribution of the detected joint coordinates. Using 6 classifiers, 3 ballet motion classifications are analyzed through 4 performance indicators. Gradient Boosting Classifier obtained the most optimal performance among the compared models. As a result of the feature importance analysis, it was confirmed that the ankle joint is important. We also analyzed the performance of the model according to four feature extraction method. And the Odds ratio was obtained and the prediction success rates for each motion were compared. The proposed method can coach ballet movements.

색인어 : 발레, 심플 포즈, 오픈 포즈, 관절 좌표 추출, 특징 중요도 분석

Keyword : Ballet, Simple pose, Open pose, Keypoints detection, Analysis of feature importance

<http://dx.doi.org/10.9728/dcs.2022.23.8.1349>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 29 June 2022; **Revised** 01 August 2022

Accepted 16 August 2022

***Corresponding Author, Seung-Ho Choi**

Tel 

E-mail: jcn99250@naver.com

I. 서론

최근 홈 트레이닝에 대한 수요가 높아짐에 따라, 집에서 발레를 즐기는 사람들이 증가했다. 발레는 정확한 자세를 취하는 것이 중요한데, 동작을 코칭 해주는 사람이 없다면 운동의 효과를 보기 어렵고 부상을 당할 위험이 있다. 일례로 발이 180°가 되는 턴 아웃 자세를 취할 때, 무릎이 바깥쪽으로 향하지 않은 상태에서 발만 180°를 이룬 채 동작을 하면, 무릎에는 전단력이 발생하게 되고 이를 알아채지 못하고 반복적으로 연습하면 무릎 인대나 골반에 부상을 입을 수 있다 [1]. 이에 사용자가 카메라 앞에서 동작을 하면 실시간으로 사용자가 어떤 자세를 한 것인지 예측하여, 정확한 자세를 위해 팔, 다리, 어깨 등이 어떻게 수정되어야 하는지 화면에 텍스트로 제시가 필요하다. 기존의 운동 코칭 서비스는 사전 정의된 루틴에 따라 동작을 하도록 시키고, 그 자세에 대해서 코칭을 하는 반면, 이 서비스는 자신의 루틴대로 자율적으로 동작을 취하면, 어떤 동작인지 예측하고 조언한다는 점에서 사용자의 자율성을 높여줄 수 있다. 따라서 본 논문에서의 공헌도는 아래와 같다.

첫 번째, 발레 동작을 분류할 수 있는 시스템을 제안한다.

두 번째, 발레 동작을 분류하는데 영향성을 끼치는 특징에 대해서 중요도 분석을 수행한다.

세 번째, 4가지 방법을 통해 추출 또는 선택한 특징만을 사용하였을 때 성능분석 실험을 한다.

II. 관련 연구

1) 자세 추정

자세를 추정하는 작업은 크게 2차원 범주와 3차원 범주로 분류한다. 2차원 범주는 추정하고자 하는 사람의 인원수에 따라 Single-person 추정 문제와 Multi-person 추정 문제로 나뉜다. 또한 Single-person 추정 문제는 접근 방법에 따라 회귀를 이용한 방식과 히트 맵을 이용하는 방식으로 나뉘며, Multi-person 추정 문제는 하향식 접근 방식과 상향식 접근 방식으로 나뉜다.

고전적인 컴퓨터 비전의 영상처리 방식은 특징을 검출하고, 검출기를 만들어 결과를 합치는 번거로운 방식인 반면, Convolution Neural Network (CNN)를 사용하면 한 번에 모든 관절을 조합하여 예측할 수 있어 매우 간단해진다. DeepPose는 CNN을 사용하여 처음으로 인상적인 성능을 달성한 모델이다 [2]. Single-person을 대상으로 하는 회귀방식으로, 큰 이미지를 사용하여 CNN 모델을 통과시켜 위치를 예측하는 Initial stage와, 이전 Stage에서 나온 예측 값 주변으로 Bounding box를 그려 이를 다시 CNN 모델에 통과시키는 N개의 Stage로 구성되어 있다.

히트 맵 방식은 각 관절이 위치할 확률을 나타내는 히트 맵을 이용하여 신체 부위를 추정하는 방식이다. N개의 관절을 예측하기 위하여 N개의 히트 맵을 추정한다. Ground truth

에 해당하는 히트 맵은 Ground truth에 해당하는 관절의 위치에 2D Gaussian을 적용하여 생성된다. 이렇게 생성된 정답 히트 맵과 예측 히트 맵 간의 Mean Squared-Error 등을 오차로 사용하여 훈련한다. 히트 맵을 이용한 방식은 위치 정보를 더 풍부하게 보존하고 CNN을 훈련시키기에 더 효과적이어서 널리 사용되는 방식이다.

3차원 범주는 하나의 시점으로 한 명을 다루는 문제, 하나의 시점으로 여러 명을 다루는 문제, 그리고 여러 시점으로 여러 명을 다루는 문제로 분류한다.

2) Open pose

관절의 위치를 정확하게 예측하기 위하여 Open pose를 사용한다 [3]. Open pose의 구조는 여러 Stage로 이루어진다. 먼저 Visual Geometry Group network-19 (VGG-19)의 Convolution layer와 Pooling layer를 통과하며 추출한 Feature map을 첫 번째 네트워크의 입력으로 넣는다. 이 네트워크는 관절의 위치를 추정하는 Confidence map을 구하는 Branch와 Part affinity fields를 구하는 Branch로 구성된다. Part affinity fields는 한 관절에서 다른 관절로 이어지는 방향으로 인코딩한 2차원 벡터이며 인체 부위 사이의 연관 정도를 나타낸다. 이를 통해 관절이 누구의 관절인지 추정할 수 있다. 첫 번째 단계에서 얻어낸 Confidence map과 Part affinity fields를 결합하여 새로운 특징을 생성하고 이를 반복적으로 다음 단계로 전달하며 네트워크를 훈련하는 구조이다.

3) Simple pose

또한 관절의 위치를 빠르게 예측하기 위하여 Simple pose를 사용한다 [4]. Simple pose는 네트워크의 복잡성을 줄여 좋은 성능을 달성한 간단한 모델이다. ResNet을 Backbone network로 사용하며, Deconvolution layer를 추가하여 High resolution feature map을 얻을 수 있다. Cascaded Pyramid Network (CPN) 또는 Hourglass 같은 기존의 네트워크들이 High resolution feature map을 얻기 위해 여러 단계에서 추출한 Feature map을 합치는 복잡한 구조를 갖는데 비해, Simple pose는 매우 간단한 구조로 비슷한 성능을 달성한다 [5], [6].

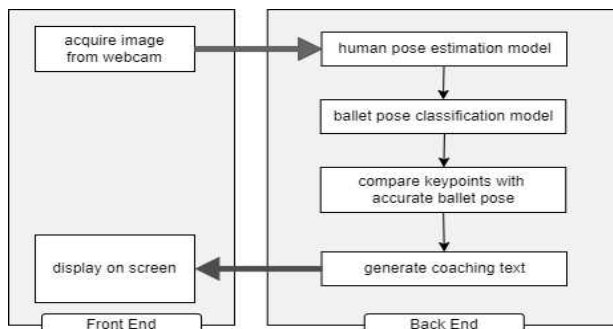


그림 1. 시스템 구성도
Fig. 1. System Configuration

Pose tracking 과정에서는 Human detector를 통해 얻은 정보와 이전 Frame의 Optical flow를 통해 생성된 정보를 합쳐 Tracking을 진행하며, Flow based pose similarity metric을 사용하여 Greedy matching을 수행한다.

III. 제안 방법

그림 1은 실시간 발레 동작 코칭 시스템의 흐름을 간략화하여 설명한다. 웹캠을 통해 실시간으로 이미지를 얻는다. 이미지는 사람의 관절의 위치를 예측하는 모델에 의해 처리되어 관절의 좌표 값으로 나타내어진다. 사용자는 Realtime mode와 Detailed mode 중 원하는 mode를 선택할 수 있다. Realtime mode는 실시간으로 코칭을 지원하며, Detailed mode는 10초 타이머를 설정하여 사용자가 원하는 자세를 코칭 받을 수 있다. Simple pose, Mobile pose, Alpha pose, MPII, 그리고 BODY-25 총 5개의 관절 검출 모델의 Frames per second (FPS)를 측정한 결과, 표 1과 같이 17개의 관절 포인트를 추정하는 Simple pose 모델이 25개의 관절을 추정하는 BODY-25 모델보다 실행 시간 측면에서 효율적이기 때문에 Realtime mode에는 Simple pose 모델을, Detailed mode에는 BODY-25 모델을 사용하도록 설계했다 [3], [4], [7]. 관절의 좌표 값은 분류 모델을 거쳐 발레 동작인 Passe, Arabesque, 그리고 Plie 중 한 클래스로 분류된다. 발레 동작 코칭 시스템을 동작 분류 문제로 정의한 이유는 발레 동작을 분류하기 위함이다.

다음으로 데이터 세트 수집에 대해 고려한다. 발레 동작은 사람의 일상적인 동작 또는 피트니스 동작과는 매우 다르기 때문에 COCO나 MPII 또는 AI Hub 등에서 제공하는 데이터 세트를 사용할 수 없다. 때문에 3가지 운동 상태에 해당하는 이미지를 직접 수집하고 레이블링하여 데이터 세트를 구축했다. 그림 2는 본 연구에서 다룬 3가지 발레 동작이다. (a)는 Passe, (b)는 Arabesque, 그리고 (c)는 Plie 동작이다. 한쪽 다리를 구부려서 발끝이 반대쪽 다리의 무릎에 닿도록 하는 동작은 Passe로 레이블링했다. 한쪽 다리 중심을 잡고 반대쪽 다리를 뒤로 뻗는 동작을 Arabesque로 레이블링했다.

표 1. 5가지 관절 검출 모델의 Frames Per Second (FPS)
Table 1. Frames Per Second (FPS) of five key points detection models

	FPS (Hz)
Simple Pose Model [4]	57
Mobile Pose Model [4]	55
Alpha Pose Model [7]	19
MPII Model [3]	1
BODY-25 Model [3]	0.376

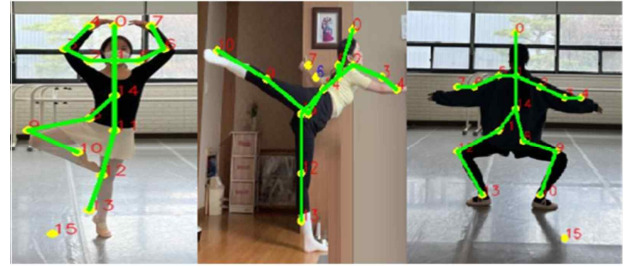


그림 2. 획득된 영상과 유효한 관절들이 시각화된 모습, (a) 파세, (b) 아라베스크, (c) 플리에

Fig. 2. Visualization of the acquired image and valid joints, (a) Passe, (b) Arabesque, (c) Plie

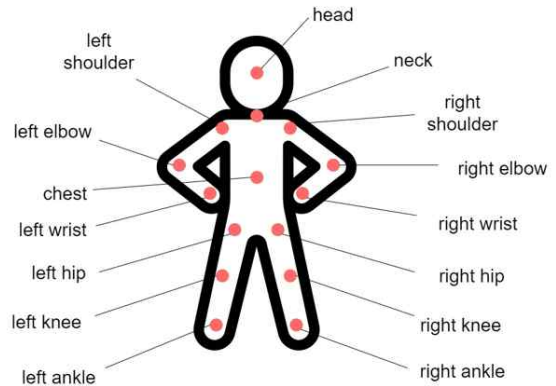


그림 3. 레이블링한 15개 관절의 종류
Fig. 3. 15 types of joints labeled

또한 양쪽 다리가 모두 지면에 닿은 채로 무릎이 바깥 방향을 향하도록 구부리는 동작을 Plie로 레이블링했다. 총 2767장의 이미지를 훈련데이터로 사용하였으며, 한 이미지 당 15개의 관절 좌표를 레이블링했다. 그림 3은 레이블링을 수행한 관절의 종류를 나타낸다.

분류 모델의 성능 검증 과정에는 총 6가지 분류기를 사용하여 실험했다. E.g. Random Forest Classifier, XGB Classifier, Gradient Boosting Classifier, Ada Gradient Boosting Classifier, Bagging Classifier, 그리고 Extra Trees Classifier [8]-[13]. 정확한 동작을 취하도록 조인하는 텍스트를 생성하기 위해서, 관절 간의 각도를 계산하여 사전 정의해 놓은 동작별 올바른 관절 간의 각도 값과 비교한다. 두 값의 오차가 설정한 역치를 초과하면 텍스트를 생성하여 화면에 표시한다. 더불어 관절의 예측 확률이 0.2 이하인 관절 좌표들을 신뢰성이 없다고 간주하여 제외한 후, 연관 관절들을 연결하여 시각화한다. 그림 2는 얻은 관절들 중 신뢰성 있는 관절을 시각화한 모습이다.

IV. 실험 결과

표 2. 각 관절 좌표 히스토그램의 다양한 분위 수 분석

Table 2. Analysis of different quantiles of the histogram of each joint coordinate

	Percentile rank (%)								
	0.05	0.5	1	99	99.5	99.95	100	10	90
head_x	0.1522	0.1903	0.2037	0.7826	0.8135	0.8719	0.8805	0.3166	0.6360
head_y	0.0	0.0416	0.0466	0.5217	0.5555	0.7130	0.7193	0.0979	0.4100
neck_x	0.1430	0.1818	0.2031	0.7905	0.8101	0.8628	0.8813	0.3157	0.6414
neck_y	0.0625	0.1142	0.1238	0.5698	0.6037	0.7515	0.7516	0.1831	0.4619
rshoulder_x	0.1628	0.1949	0.2226	0.8422	0.8606	0.9096	0.9346	0.3560	0.7052
rshoulder_y	0.1073	0.1358	0.1490	0.6033	0.6213	0.7620	0.7670	0.2192	0.4927
relbow_x	0.1325	0.1872	0.2148	0.9099	0.9239	0.9514	0.9557	0.3575	0.8025
relbow_y	0.0950	0.1330	0.1421	0.6202	0.6479	0.7643	0.7796	0.2039	0.5086
rwrist_x	0.0305	0.0853	0.1184	0.9579	0.9664	0.9885	0.9980	0.3008	0.8943
rwrist_y	0.0318	0.0440	0.0552	0.6723	0.7172	0.7975	0.8063	0.1295	0.5323
lshoulder_x	0.1030	0.1400	0.1563	0.7541	0.7944	0.8408	0.8515	0.2508	0.6038
lshoudler_y	0.1038	0.1356	0.1477	0.5948	0.6237	0.7624	0.7778	0.2162	0.4929
lelbow_x	0.0569	0.0762	0.0933	0.7860	0.8203	0.9745	0.9860	0.1687	0.5835
lelbow_y	0.0912	0.1273	0.1375	0.6411	0.6569	0.7731	0.7853	0.2013	0.5067
lwrist_x	0.0088	0.0262	0.04	0.8673	0.8937	0.9342	0.9428	0.1012	0.6153
lwrist_y	0.0214	0.0472	0.0593	0.6650	0.7265	0.8137	0.8286	0.1297	0.5357
rhip_x	0.1934	0.2415	0.2606	0.8126	0.8391	0.8903	0.8956	0.3952	0.6443
rhip_y	0.2457	0.2977	0.3146	0.7457	0.7687	0.8500	0.8647	0.4032	0.6470
rknee_x	0.0880	0.2619	0.3127	0.9186	0.9407	0.9766	0.9805	0.4251	0.7969
rknee_y	0.2264	0.2944	0.3108	0.8139	0.8305	0.8573	0.8745	0.4117	0.7328
rankle_x	0.08	0.2335	0.3049	0.9292	0.9437	0.9645	0.9712	0.4097	0.8490
rankle_y	0.1050	0.2137	0.2444	0.9673	0.9809	0.9929	0.9929	0.4217	0.9052
lhip_x	0.0893	0.1644	0.1878	0.7564	0.7859	0.8423	0.8663	0.3158	0.5610
lhip_y	0.2410	0.2974	0.3094	0.7470	0.7716	0.8480	0.8664	0.4118	0.6514
lknee_x	0.0325	0.0657	0.0824	0.7205	0.7781	0.9458	0.9458	0.1594	0.5502
lknee_y	0.2469	0.2884	0.3105	0.8263	0.8399	0.8646	0.8729	0.4345	0.7461
lankle_x	0.0123	0.0304	0.0362	0.8072	0.8465	0.8843	0.9402	0.1513	0.5683
lankle_y	0.1451	0.2249	0.2451	0.9763	0.9876	0.9993	0.9993	0.4835	0.9208
chest_x	0.1399	0.1808	0.2051	0.7872	0.8073	0.8576	0.8813	0.3259	0.6331
chest_y	0.1838	0.2246	0.2407	0.6771	0.7027	0.8084	0.8198	0.3272	0.5768

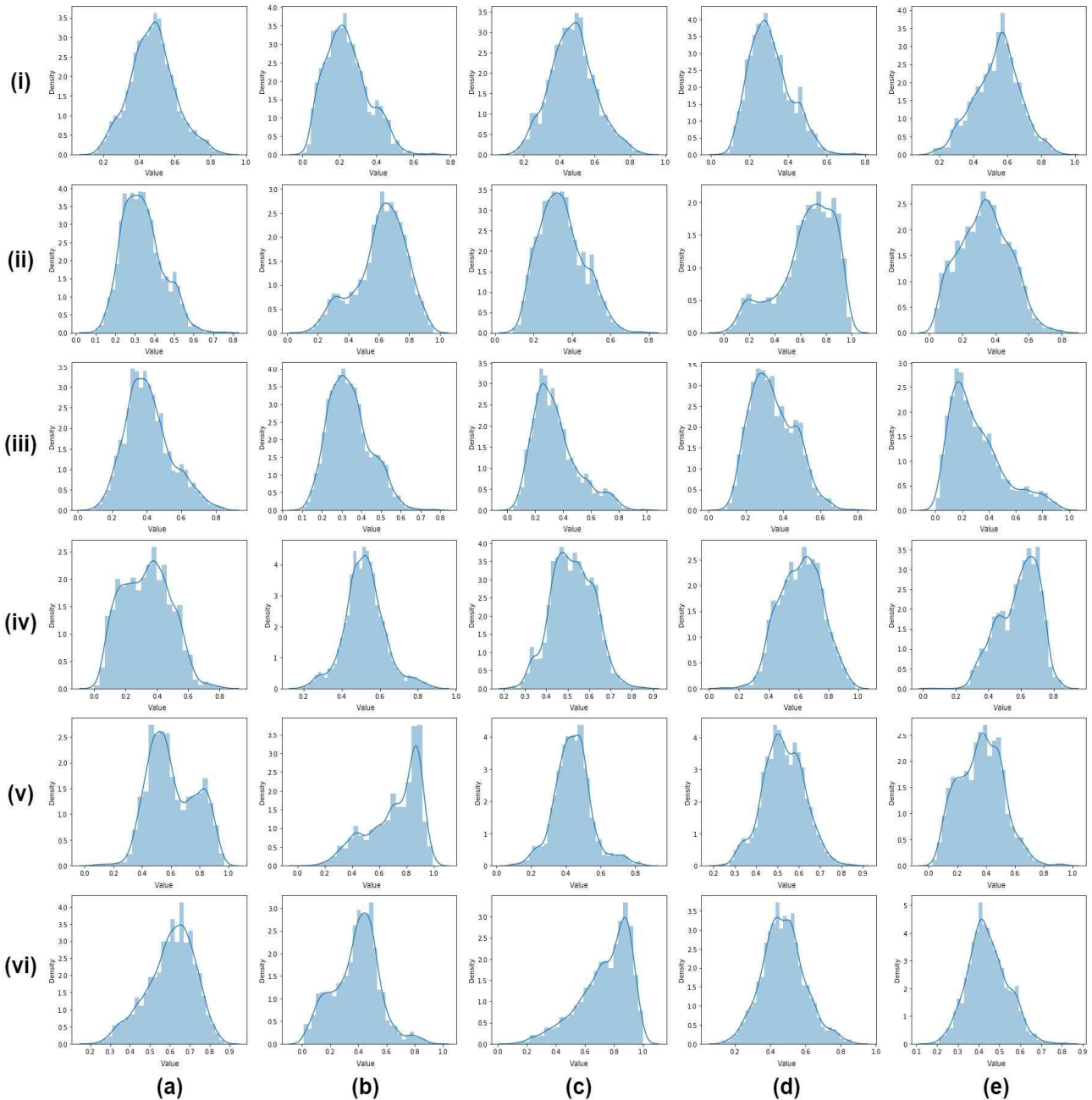


그림 4. 15가지 관절의 x, y 좌표에 대한 히스토그램 (i).(a) head_x, (i).(b) head_y, (i).(c) neck_x, (i).(d) neck_y, (i).(e) rshoulder_x, (ii).(a) rshoulder_y, (ii).(b) relbow_x, (ii).(c) relbow_y, (ii).(d) rwrist_x, (ii).(e) rwrist_y, (iii).(a) lshoulder_x, (iii).(b) lshoulder_y, (iii).(c) lelbow_x, (iii).(d) lelbow_y, (iii).(e) lwrist_x, (iv).(a) lwrist_y, (iv).(b) rhip_x, (iv).(c) rhip_y, (iv).(d) rknee_x, (iv).(e) rknee_y, (v).(a) rankle_x, (v).(b) rankle_y, (v).(c) lhip_x, (v).(d) lhip_y, (v).(e) lknee_x, (vi).(a) lknee_y, (vi).(b) lanklex_x, (vi).(c) lankle_y, (vi).(d) chest_x, (vi).(e) chest_y

Fig. 4. Histogram of x and y coordinates of 15 joints. (i).(a) head_x, (i).(b) head_y, (i).(c) neck_x, (i).(d) neck_y, (i).(e) rshoulder_x, (ii).(a) rshoulder_y, (ii).(b) relbow_x, (ii).(c) relbow_y, (ii).(d) rwrist_x, (ii).(e) rwrist_y, (iii).(a) lshoulder_x, (iii).(b) lshoulder_y, (iii).(c) lelbow_x, (iii).(d) lelbow_y, (iii).(e) lwrist_x, (iv).(a) lwrist_y, (iv).(b) rhip_x, (iv).(c) rhip_y, (iv).(d) rknee_x, (iv).(e) rknee_y, (v).(a) rankle_x, (v).(b) rankle_y, (v).(c) lhip_x, (v).(d) lhip_y, (v).(e) lknee_x, (vi).(a) lknee_y, (vi).(b) lanklex_x, (vi).(c) lankle_y, (vi).(d) chest_x, (vi).(e) chest_y

1) 관절 데이터 분포 확인

관절 검출 모델에서 추출한 15가지 관절의 X, Y좌표를 분 류모델의 입력으로 사용한다. 이에 앞서 30개 Feature의 분

포를 확인했다. 30개 Feature는 딥러닝 모델을 통과하면서 0 과 1사이로 스케일링이 되어있는 상태이다. 그림 4는 각 관절 좌표의 히스토그램이다.

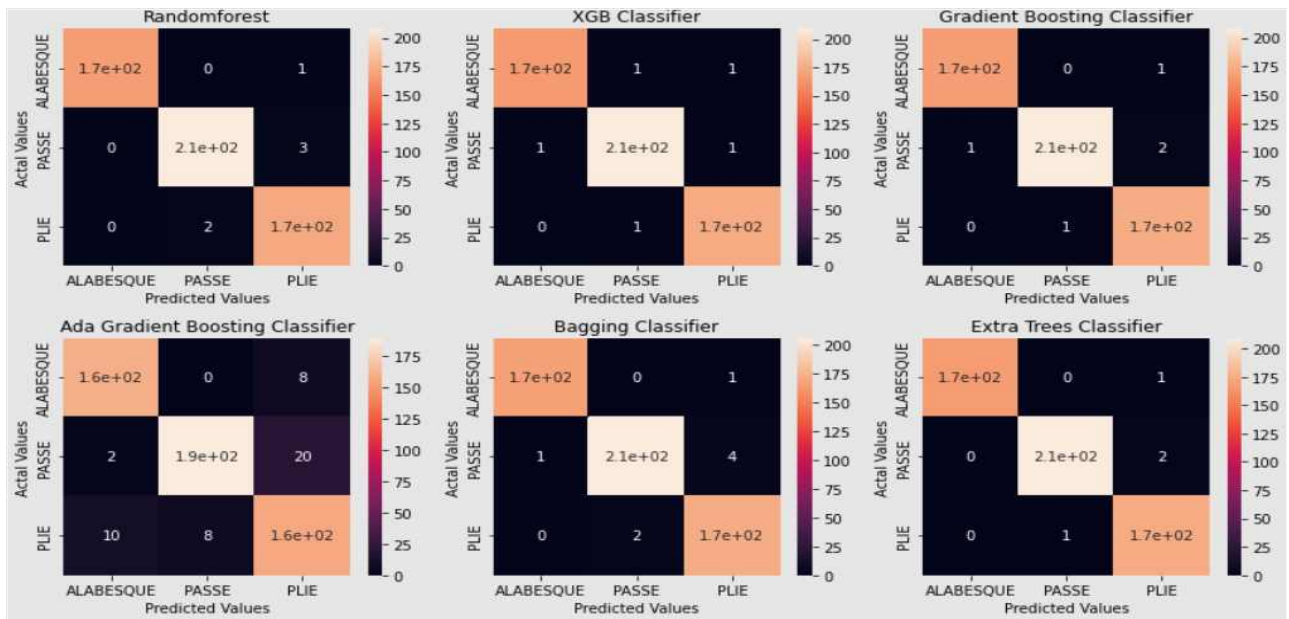


그림 5. 6가지 발레 동작 분류 모델의 Confusion matrix
 Fig. 5. Confusion matrix of six ballet action classification models

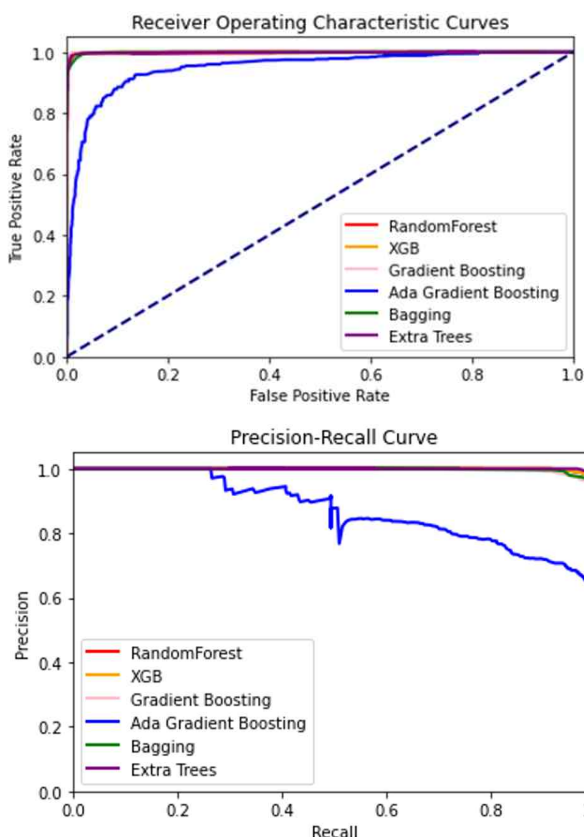


그림 6. 6가지 발레 동작 분류 모델의 Receiver Operating Characteristic Curves (ROC Curves)와 Precision-Recall Curves (PR Curves)

Fig. 6. Receiver Operating Characteristic Curves (ROC Curves) and Precision-Recall Curves (PR Curves) of six ballet action classification models

표 3. 6가지 발레 동작 분류 모델의 정밀도, 재현율, F1 점수, 그리고 Area of Under the Curve (AUC) 점수

Table 3. Precision, Recall, F1 score, and Area of Under the Curve (AUC) score of six ballet action classification models

	Precision	Recall	F1 score	AUC score
Random Forest	0.976	0.977	0.977	0.998
XGB Classifier	0.993	0.992	0.992	0.999
Gradient Boosting Classifier	0.990	0.988	0.989	0.999
Ada Gradient Boosting Classifier	0.912	0.912	0.912	0.960
Bagging Classifier	0.979	0.978	0.978	0.996
Extra Trees Classifier	0.991	0.990	0.991	0.999

표 4. 3가지 발레 동작의 Odds 분석

Table 4. Analysis of odds of ballet three class

	Odds
Passe	70.42
Arabesque	165.67
Plie	82.33

두드러지는 특이치는 존재하지 않았으며 lwrist_x, rankle_y, 그리고 lankle_y는 한쪽에 치우친 모양을 가짐을 확인했다. 표 2는 각 관절 좌표의 히스토그램에 대한 0.05%, 0.5%, 1%, 99%, 99.5%, 그리고 100%에 해당하는 값과 10% 하한, 10% 상한에 해당하는 값을 나타낸다. 오른쪽 발목 관절의 X 좌표인 rankle_x는 왼쪽 발목 관절의 X 좌표인 lankle_x 보다 큰 값을 가지며, 오른쪽 발목 관절의 Y 좌표인 rankle_y와 왼쪽 발목 관절의 Y 좌표인 lankle_y는 비슷한 값을 가졌다. 이는 발목관절의 X 좌표는 왼쪽, 오른쪽에 따라 특정 값을 가지는 경향이 있지만, Y 좌표는 왼쪽, 오른쪽에 상관없이 다양한 값을 가지는 것으로 해석할 수 있다.

2) 발레 동작 분류 모델의 성능 비교

발레 동작 분류 모델의 성능을 비교하기 위하여 Random Forest Classifier, XGB Classifier, Gradient Boosting Classifier, Ada Gradient Boosting Classifier, Bagging Classifier, 그리고 Extra Trees Classifier를 사용했다. 표 3은 각 모델의 Precision, Recall, F1 score, AUC score 값을 나타내며 XGB Classifier의 Precision, Recall, F1 score 성능이 가장 좋음을 확인했다. 그림 5는 각 모델의 Confusion Matrix를 나타낸다.

Ada Gradient Boosting Classifier를 제외한 5가지 모델은 동작을 잘 분류함을 확인할 수 있다.

그림 6은 각각 모델의 ROC Curve와 PR Curve를 나타낸다. 6가지 분류 모델 중 Ada Gradient Boosting 모델의 성능이 가장 낮았다.

3) 동작의 승산 비 분석

표 4는 Passe, Arabesque, 그리고 Plie에 동작에 대한 Odds를 나타낸다. 이를 기반으로 각 동작에 대한 Odds ratio를 구할 수 있다. Arabesque에 대한 Passe의 Odds ratio는 0.425, Plie에 대한 Passe의 Odds ratio는 0.855이다. 즉 Passe 동작의 예측 성공확률은 Arabesque 동작의 예측 성공확률의 0.425배이고, Plie 동작의 0.885 배이다. Passe에 대한 Arabesque의 Odds ratio는 2.353, Plie에 대한 Arabesque의 Odds ratio는 2.012이다. 즉 Arabesque 동작의 예측 성공 확률은 Passe 동작의 예측 성공 확률의 2.353배이고, Plie 동작의 2.012배이다. Passe에 대한 Plie의 Odds ratio는 1.169, Arabesque에 대한 Plie의 Odds ratio는 0.497이다. 즉 Plie 동작의 예측 성공 확률은 Passe 동작의 1.169배이고, Arabesque 동작의 0.497배이다.

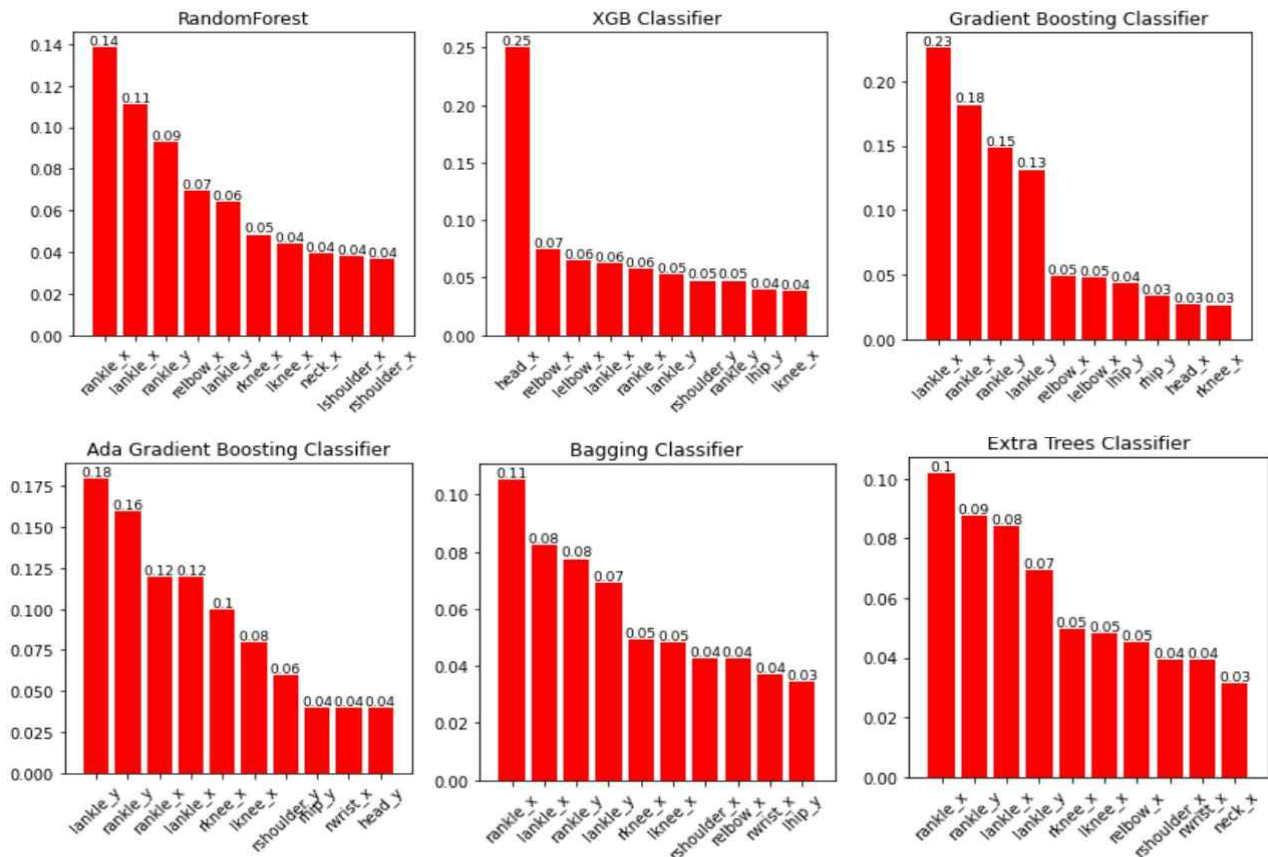


그림 7. 6가지 발레 행동 분류 모델에서 Feature importance 가 높은 상위 10가지 특징
 Fig. 7. Top 10 features with high feature importance in six ballet action classification models

표 5. 4가지 특징 추출 또는 선택 방법의 6가지 모델에 대한 성능 비교

Table 5. Comparison of four feature extraction or selection methods for six models

feature extraction and selection method	Model	Top 5 features				Top 10 features				Top 15 features			
		Precision	Recall	F1 score	AUC score	Precision	Recall	F1 score	AUC score	Precision	Recall	F1 score	AUC score
Model based feature selection	Random forest Classifier	0.994	0.994	0.994	0.999	0.987	0.987	0.987	0.999	0.992	0.992	0.992	0.999
	XGB Classifier	0.901	0.894	0.897	0.981	0.983	0.984	0.984	0.999	0.993	0.993	0.993	0.999
	Gradient Boosting Classifier	0.992	0.992	0.992	0.999	0.985	0.986	0.985	0.999	0.994	0.994	0.994	0.999
	Ada Gradient Boosting Classifier	0.843	0.840	0.841	0.904	0.883	0.881	0.881	0.945	0.896	0.894	0.894	0.943
	Bagging Classifier	0.985	0.985	0.985	0.999	0.972	0.971	0.972	0.997	0.975	0.975	0.975	0.998
	Extra Trees Classifier	1.0	1.0	1.0	1.0	0.992	0.993	0.992	0.999	0.998	0.998	0.998	1.0
Principal component feature extraction	Random forest Classifier	0.943	0.940	0.941	0.989	0.979	0.978	0.978	0.999	0.996	0.996	0.996	0.999
	XGB Classifier	0.907	0.905	0.905	0.980	0.978	0.975	0.976	0.999	0.989	0.989	0.989	0.999
	Gradient Boosting Classifier	0.917	0.916	0.916	0.984	0.977	0.976	0.976	0.999	0.992	0.993	0.992	0.999
	Ada Gradient Boosting Classifier	0.802	0.801	0.800	0.881	0.884	0.878	0.880	0.950	0.966	0.965	0.965	0.983
	Bagging Classifier	0.909	0.908	0.909	0.974	0.967	0.965	0.966	0.993	0.947	0.948	0.947	0.991
	Extra Trees Classifier	0.956	0.955	0.955	0.994	0.992	0.993	0.992	0.999	0.996	0.996	0.996	0.999
Independent component feature extraction	Random forest Classifier	0.947	0.945	0.946	0.991	0.973	0.972	0.973	0.998	0.989	0.989	0.989	0.999
	XGB Classifier	0.905	0.900	0.902	0.982	0.966	0.965	0.965	0.997	0.991	0.990	0.991	0.999
	Gradient Boosting Classifier	0.927	0.926	0.926	0.985	0.974	0.972	0.973	0.998	0.992	0.993	0.992	0.999
	Ada Gradient Boosting Classifier	0.816	0.812	0.814	0.867	0.933	0.931	0.932	0.980	0.946	0.935	0.940	0.953
	Bagging Classifier	0.934	0.930	0.932	0.985	0.952	0.951	0.951	0.989	0.964	0.967	0.966	0.997
	Extra Trees Classifier	0.959	0.958	0.958	0.993	0.993	0.992	0.992	0.999	0.998	0.998	0.998	0.999
Univariate feature selection	Random forest Classifier	0.930	0.926	0.928	0.987	0.981	0.980	0.980	0.999	0.988	0.990	0.989	0.999
	XGB Classifier	0.900	0.891	0.894	0.982	0.985	0.985	0.985	0.999	0.996	0.996	0.996	0.999
	Gradient Boosting Classifier	0.909	0.899	0.902	0.983	0.985	0.986	0.985	0.999	0.996	0.996	0.996	0.999
	Ada Gradient Boosting Classifier	0.718	0.718	0.700	0.828	0.878	0.871	0.871	0.937	0.930	0.924	0.925	0.959
	Bagging Classifier	0.923	0.918	0.920	0.979	0.969	0.968	0.969	0.994	0.982	0.982	0.982	0.994
	Extra Trees Classifier	0.939	0.935	0.936	0.989	0.986	0.985	0.985	0.999	0.994	0.995	0.994	0.999
Average of feature extraction and selection	Random forest Classifier	0.953	0.951	0.952	0.991	0.980	0.979	0.979	0.998	0.991	0.991	0.991	0.999
	XGB Classifier	0.903	0.897	0.899	0.981	0.978	0.977	0.977	0.998	0.992	0.992	0.992	0.999
	Gradient Boosting Classifier	0.936	0.933	0.934	0.987	0.980	0.980	0.979	0.998	0.993	0.994	0.993	0.999
	Ada Gradient Boosting Classifier	0.794	0.792	0.788	0.870	0.894	0.890	0.891	0.953	0.934	0.929	0.931	0.959
	Bagging Classifier	0.937	0.935	0.936	0.984	0.965	0.963	0.964	0.993	0.967	0.968	0.967	0.995
	Extra Trees Classifier	0.963	0.962	0.962	0.994	0.990	0.990	0.990	0.999	0.996	0.996	0.996	0.999

4) 특징의 중요도 분석

그림 7은 각 모델이 예측과정에서 중요하게 사용한 상위 10개 Feature의 Feature importance를 나타내며, 6개 중 5개의 모델에서 발목 관절이 가장 중요하게 사용된데 비하여, XGB Classifier에서는 발목 관절이 다른 관절들과 비슷한 중요도로 사용됨을 확인했다.

또한 Shap을 이용하여 발레 동작의 클래스를 결정하는데 중요하게 사용된 특징과 중요도를 구하고, 6가지 모델의 평균을 계산하여 그림 8에 나타내었다. 결과적으로 상위 4가지 특징이 모두 발목과 관련된 특징으로 나타났다. 따라서 발목 관절이 가장 중요하게 사용되었음을 확인할 수 있다. Class 0에 해당하는 Passe 동작을 결정하는데 rankle_y와 lankle_y가 높은 비중을 차지했다. 이는 한쪽 다리를 구부리는 Passe 동작의 특징이 잘 반영된 것으로 이해할 수 있다. 또한 Class 1에 해당하는 Arabesque 동작을 결정하는데 rankle_x와 lankle_x가 높은 비중을 차지했다. 이는 한쪽 다리를 멀리 뻗는 해당 동작의 특징이 잘 반영된 것으로 이해할 수 있다. Class 3에 해당하는 Passe 동작은 두 발 모두 땅에 닿아야 하는 해당 동작을 결정짓기 위하여 rankle_x, rankle_y, lankle_x, 그리고 lankle_y가 모두 비슷한 비율로 사용되었음을 확인할 수 있다.

다음으로 모델의 성능을 높이기 위해 중요한 특징만을 사용하여 모델을 구성했다. Model based feature selection, Principal component feature extraction, Independent component feature extraction, 그리고 Univariate feature selection 방법을 통해 선택 또는 추출한 상위 5가지, 10가지, 15가지의 특징만을 사용해 Precision, Recall, F1 score, AUC score를 측정하여 표 5에 나타냈다. 4가지 분석 방법에 대해 평균적으로 상위 15개, 10개, 그리고 5개 특징만 사용한 경우의 순서로 모델의 성능이 높았다. 전체 특징을 사용했을 때와 비교했을 때, 평균적으로 상위 15개 특징만 사용한 경우 성능이 향상되었다. 하지만 상위 5개 특징만 사용한 경우 모델의 성능이 하락했다. 이를 통해 평균적으로 상위 15개 특징을 사용하는 것이 상위 5개 특징을 사용하는 것 보다 모델의 학습에 도움이 됨을 확인했다. 또한 4가지 특징 분석 방법을 6가지 모델에 대한 평균 F1 score를 기준으로 비교하였을 때, Model based feature selection, Independent component feature extraction, Principal component feature extraction, 그리고 Univariate feature selection 방법 순서로 F1 score가 높았다. 이를 통해 평균적으로 Model based feature selection 방법이 6가지 모델에 대해 데이터를 잘 표현하는 특징을 선택해줌을 확인했다. 전체 성능 지표에서 가장 높은 성능을 달성한 모델은 Extra Trees Classifier이며, Model based feature selection 방법을 통해 추출한 상위 5개의 특징만 사용하여 학습한 경우이다. 따라서 전체 특징을 사용하여 학습하는 것 보다 Model based feature selection 방법을 통해 중요한 특징 5가지를 선택하여 학습하는 것이 가장 효과적임을 확인했다.

5) 발레 동작 분류 모델의 FPS 측정

표 6은 6가지 분류 모델의 1초당 처리하는 프레임의 수를 측정하여 나타낸다. Gradient Boosting Classifier가 실행 속도 측면에서 가장 효율적임을 확인했다. 앞의 Precision, Recall, F1 score, 그리고 AUC score에서 가장 높은 성능을 달성한 Extreme Trees Classifier는 가장 느렸다. 정확도와 실행속도를 모두 고려하였을 때 행동 분류 모델로서 Gradient Boosting Classifier가 가장 적합하다고 판단했다. 실제 데모에서는 ResNet 기반의 Simple pose와 Open pose의 BODY-25 모델에서 특징을 추출하고 Gradient Boosting Classifier를 사용하여 Passe, Ara-besque, 그리고 Plie 중 어느 동작에 해당하는지 분류한다.

6) 동작 교정 대상 판단

다음으로 사전 정의된 올바른 동작과 입력된 동작의 오차를 계산하여 동작 교정 텍스트를 생성한다. Passe 동작의 경우 골반, 무릎, 그리고 발목 관절이 이루는 각도가 45° 보다 크면 자세 교정 대상으로 판단하도록 정의한다. Arabesque 동작의 경우 무릎, 엉덩이, 그리고 반대쪽 무릎이 이루는 각도가 90° 보다 작으면 동작 교정 대상으로 판단하도록 정의한다.

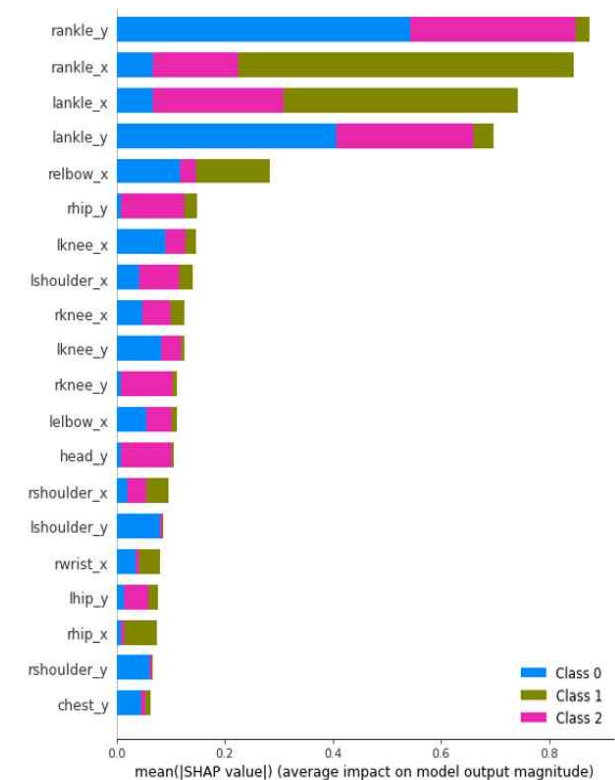


그림 8. Shap을 이용한 6가지 발레 동작 분류 모델의 Feature importance의 평균 분석 [14]

Fig. 8. Analysis of mean of feature importance in six ballet action classification models using shap [14]

표 6. 6가지 발레 동작 분류 모델의 Frames Per Second

Table 6. Frames Per Second of ballet action six classification models

Model	FPS (Hz)
RandomForest Classifier	42
XGB Classifier	227
Gradient Boosting Classifier	312
Ada Gradient Boosting Classifier	98
Bagging Classifier	109
Extra Trees Classifier	40



그림 9. Gradient Boosting Classifier를 이용한 데모 결과
Fig. 9. Demo results using Gradient Boosting Classifier

또한 Plie 동작의 경우 양쪽 다리의 엉덩이, 무릎, 그리고 발목 관절이 이루는 각도가 100° 보다 크면 동작 교정 대상으로 판단하도록 정의한다. 동작이 교정 대상이면 “Raise left leg 13° more”와 같이 각도의 오차가 0이 되도록 텍스트를 생성하고, 동작이 교정 대상이 아니라면 “Good” 텍스트를 생성한다. 최종적으로, 그림 9와 같이 생성된 텍스트와 검출된 좌표 값을 화면에 시각화한다.

V. 결 론

동작을 코칭 해주는 사람이 없이 혼자 발레를 할 때 정확한 자세를 취하기 어려운 문제점을 해소하기 위해, 관절의 위치를 인식하여 이를 통해 동작을 예측하고 현재 동작과 정확한 동작의 오차를 계산하여 코칭 텍스트를 생성하는 시스템을 제안했다. 6가지 분류기의 Precision, Recall, F1 score, 그리고 AUC score를 비교하여 가장 성능이 좋은 분류기를 확인했다. 그리고 중요하게 사용된 특징을 분석함으로써 발레 동작을 분류하는데 있어 가장 영향성이 있는 특징이 무엇인지 파악했다. 모델의 성능 향상을 위해 Model based feature selection, Principal component feature extraction, Independent component feature extraction, 그리고

Univariate feature selection을 적용하여 Top 5, Top 10, 그리고 Top 15 특징만을 뽑아 분석을 진행했다. Model based feature selection 방법을 통해 중요한 특징 5가지를 선택하여 학습하는 것이 더 효과적임을 확인했다. 그리고 각 동작 분류 모델의 FPS와 성능을 분석해 Gradient Boosting Classifier 모델이 발레 동작 분류에 적합함을 확인했다. 전체 발레 동작의 특징의 중요도를 Shap을 이용해 분석한 결과 5가지의 특징이 데이터의 중요한 부분을 차지하고 있음을 확인했다. 추후에는 Passe, Arabesque, 그리고 Plie 동작뿐만 아니라 더 많은 동작을 다뤄보고자 한다.

참고문헌

- [1] M. J. Kim, I. J. Lee, and S. Y. Lee, "The differences of static lower extremity alignment between female ballet students and female non-ballet students", *The Korean Journal of Physical Education*, Vol. 60, No. 3, pp. 405-413, May 2022. <https://doi.org/10.23949/kjpe.2021.5.60.3.28>
- [2] A. Toshev, and C. Szegedy, "DeepPose: Human pose estimation via deep neural networks.", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Columbus, pp. 1653-1660, August 2014. <https://doi.org/10.1109/CVPR.2014.214>
- [3] Z. Cao, T. Simon, S. E. Wei, and Y. Sheikh, "Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, pp. 7291-7299, April 2017. <https://doi.org/10.1109/CVPR.2017.143>
- [4] B. Xiao, H. Wu, and Y. Wei, "Simple baselines for human pose estimation and tracking", in *Proceedings of the European conference on computer vision (ECCV)*, Munich, pp. 466-481, August 2018. <https://doi.org/10.48550/arXiv.1804.06208>
- [5] Y. Chen, Z. Wang, Y. Peng, Z. Zhang, G. Yu, and J. Sun, "Cascaded pyramid network for multi-person pose estimation", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Salt Lake City, pp. 7103-7112, April 2018. <https://doi.org/10.1109/CVPR.2018.00742>
- [6] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation", in *European conference on computer vision*, Amsterdam, pp. 483-499, September 2016. https://doi.org/10.1007/978-3-319-46484-8_29
- [7] H. S. Fang, S. Xie, Y. W. Tai, and C. Lu, "Rmpe: Regional multi-person pose estimation", in *Proceedings of the IEEE*

international conference on computer vision, Venice, pp. 2334-2343, October 2017.
<https://doi.org/10.48550/arXiv.1612.00137>

[8] B. Leo, "Random forests", *Machine learning*, Vol. 45, No. 1, pp. 5-32, October 2001.
<https://doi.org/10.1023/A:1010933404324>

[9] T. Chen, and C. Guestrin, "Xgboost: A scalable tree boosting system", in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, New York, pp. 785-794, August 2016.
<https://doi.org/10.1145/2939672.2939785>

[10] J. H. Friedman, "Greedy function approximation: a gradient boosting machine", *Annals of statistics*, Vol. 29, No. 5, pp.1189-1232, October 2001.
<https://www.jstor.org/stable/2699986>

[11] Y. Freund, and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting", *Journal of computer and system sciences*, Vol. 55, No. 1, pp. 119-139, August 1997.
<https://doi.org/10.1006/jcss.1997.1504>

[12] B. Leo, "Bagging predictors", *Machine learning*, Vol. 24, No. 2, pp. 123-140, August 1996.
<https://doi.org/10.1007/BF00058655>

[13] G. Pierre, D. Ernst, and L. Wehenekel, "Extremely randomized trees", *Machine learning*, Vol. 63, No. 1, pp. 3-42, March 2006.
<https://doi.org/10.1007/s10994-006-6226-1>

[14] Shapley additive explanations [Internet]. Available:
<https://shap.readthedocs.io/en/latest/index.html>

손예진 (Yea-Jin Shon)



2022년 : 숭실대학교 기계공학부 학부과정

최승호 (Seoung-Ho Choi)



2018년 : 한성대학교 전자정보공학과 (공학사)
2020년 : 한성대학교 대학원 (공학석사)

2021년~현 재: 한성대학교 기초교양학부 시간강사
2022년~현 재: 숭실대학교 컴퓨터공학과 시간강사
2022년~현 재: 광운대학교 SW중심대학사업단 초빙교수
※ 관심분야 : 딥러닝, 컴퓨터 비전등