IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

2022년 8월 28일 접수, 2022년 9월 3일 승인, 2022년 9월 6일 발행, 2022년 9월 15일 현재 버전 발행.

디지털 객체 식별자 10.1109/ACCESS.2022.3204755

저화질 비디오에서 동작 인식 모델의 성능 평가

오타니 아오이1 , 하시구치 료타1 , 오미 카즈키1 , 후쿠시마 노리시게1 , (회원, IEEE) 및 타마키 토루1 , (회원, IEEE)

[1]나고야시 쇼와구 고키소쵸 나고야공업대학 466-8555, 나고야, 일본

교신저자: 토루 타마키(이메일: tamaki.toru@nitech.ac.jp).

이 작업은 JSPS KAKENHI 보조금 번호 JP22K12090의 일부 지원을 받았습니다.

초록

동작 인식 모델을 설계할 때 동영상의 품질은 중요한 문제이지만, 품질과 성능 간의 균형은 종종 무시됩니다. 일반적으로 동작 인식 모델은 고품질 동영상으로 학습되기 때문에 저화질 동영상으로 테스트할 때 모델 성능이 어떻게 저하되는지, 학습 동영상의 품질이 성능에 어느 정도 영향을 미치는지 알 수 없습니다. 비디오 품질 문제는 중요하지만 지금까지 연구된 바가 없습니다. 본 연구의 목표는 다양한 화질의 트랜스코딩된 비디오에 대한 여러 동작 인식 모델의 정량적 성능 평가를 통해 학습 및 테스트 비디오의 성능과 화질 간의 상충 관계를 보여주는 것입니다. 먼저, 비디오 품질이 사전 학습된 모델의 성능에 어떤 영향을 미치는지 보여줍니다. JPEG(압축 강도)와 H.264/AVC(CRF)의 품질 관리 파라미터를 변경하여 Kinetics400의 원본 검증 영상을 트랜스코딩합니다. 그런 다음 트랜스코딩된 비디오를 사용하여 사전 학습된 모델을 검증합니다. 둘째, 트랜스코딩된 비디오에서 학습된 모델의 성능을 보여줍니다. JPEG와 H.264/AVC의 품질 파라미터를 변경하여 Kinetics400의 원본 트레이닝 비디오를 트랜스코딩합니다. 그런 다음 트랜스코딩된 훈련 비디오로 모델을 훈련하고 원본 및 트랜스코딩된 검증 비디오로 모델을 검증합니다. JPEG 트랜스코딩 실험 결과, 압축 강도가 70 미만인 경우 심각한 성능 저하(최대 -1.5%)가 나타나지 않아 육안으로 품질 저하가 관찰되지 않으며, 80 이상에서는 품질 지수에 따라 선형적으로 성능이 저하되는 것으로 나타났습니다. H.264/AVC 트랜스코딩 실험 결과, 동영상 파일의 총 크기가 30%로 줄어드는 동안 CRF30에서는 성능 손실(최대 -1%)이 크지 않은 것으로 나타났습니다. 요약하자면, 화질 저하가 심각하고 눈에 띄지 않는 한 비디오 품질은 동작 인식 모델의 성능에 큰 영향을 미치지 않습니다. 이를 통해 훈련 및 검증 비디오를 트랜스코딩하고 파일 크기를 원본 비디오의 3분의 1로 줄일 수 있습니다.

인덱스 용어 동작 인식, 비디오 품질, 트랜스코딩, JPEG, H.264/ACV, FFmpeg

I. 소개

최근 인공지능에 의한 영상 분석을 통해 사람의 행동을 이해하는 기술이 주목받고 있습니다[1]-[4]. 이를 위해서는 걷거나 뛰는 등 영상 속 사람의 행동을 인식하는 것이 필요한데, 이러한 작업을 *행동 인식이라고 합니다*. 대규모 데이터 세트의 등장[5]-[13]과 딥러닝 기술의 발전으로 엄청난 양의 동작 인식 방법이 제안되었습니다[14]-[18]. 이러한 기술은 현재 이상 행동 감지, 근로자 기술 평가, 스포츠 훈련 지원 등 다양한 분야에서 사용됩니다.

동작 인식을 위해서는 장면 속 사람과 사물 간의 상호작용 등 다양한 상황을 이해해야 합니다. 하지만 동영상 액션 인식에는 어려운 문제가 많고, 동영상 콘텐츠도 다양해 여러 가지 문제를 내포하고 있습니다. 예를 들어 다음과 같은 요인들이 동작 인식을 어렵게 만듭니다: 사람의 크기와

(a) (b) (c)

그림 1: 이미지 품질 저하. (a) 원본 이미지.
(b) 저해상도 이미지. (c) 저화질 이미지.

모양, 시점, 오클루전, 조명, 그림자, 데이터 크기 등을 설정할 수 있습니다.

본 연구에서는 이러한 문제점 중 비디오 인코딩 품질이 동작 인식 성능에 미치는 영향을 조사합니다. 기존의 동작 인식 모델은 저화질 동영상용으로 설계되지 않았기 때문에 저화질 동영상에서 어떤 성능을 발휘하는지는 명확하지 않습니다. 저화질 동영상이 사용되는 잠재적 상황은 다음과 같은 두 가지가 있습니다.

첫째, 동작 인식의 실제 적용 사례를 살펴보면(그림 1(a)), 입력 비디오의 품질이 높지 않을 수 있습니다. 오늘날 연구에 일반적으로 사용되는 동작 인식 데이터 세트에서 제공하는 비디오는 일반적으로 고해상도 및 고품질입니다. 그러나 카메라가 원거리에서 동작 장면을 캡처하고 관심 영역(ROI)이 잘린 경우 이미지의 해상도가 작을 수 있습니다(그림 1(b)). 또한 카메라가 대역폭이 제한된 통신망을 통해 비디오를 전송하는 경우 비디오 품질이 상당히 저하될 수 있습니다(그림 1(c)). 이러한 상황은 통신 조건에 따라 비트 레이트가 동적으로 조정되는 스트리밍에서도 발생합니다. 최근에는 사전 학습된 모델을 사용하는 것이 많은 애플리케이션에서 일반화되었지만, 이러한 저화질 비디오에서 모델이 어떻게 작동하는지는 명확하지 않습니다.

둘째, 최근 몇 년 동안 비디오 데이터 세트의 크기가 점점 커지면서 필요한 디스크 공간도 증가하고 있습니다. 동작 인식 모델을 학습시킬 때, 각 영상의 프레임을 미리 번호가 매겨진 JPEG 이미지 시퀀스로 저장하고 학습 중에 해당 JPEG 이미지를 로드하는 것이 일반적입니다. 또한, 훈련을 위한 일반적인 사전 처리는 가로 세로 비율을 유지하면서 이미지의 짧은 면을 256픽셀과 320픽셀 범위에서 크기를 조정한 다음 224×224픽셀의 패치를 임의로 자르는 것입니다. 따라서 동영상(또는 JPEG 이미지)은 일반적으로 공간을 절약하기 위해 크기가 조정됩니다. 비디오 또는 이미지 파일을 높은 압축률로 추가로 트랜스코딩하면 디스크 공간을 더 줄일 수 있으며 비디오 및 이미지 파일을 로딩하는 속도가 빨라집니다. 파일 크기가 작을수록 로컬 네트워크나 클라우드를 통해 파일을 더 빠르게 로드할 수 있기 때문에 속도 계수는 필수적입니다. 그러나 이러한 고압축 트랜스코딩은 품질 저하를 유발하고 모델 성능에 영향을 미칠 수 있으므로 일반적으로 수행하지 않습니다.

동작 인식 모델이 입력 비디오의 품질에 어떤 영향을 받는지에 대한 의문은 여전히 남아 있습니다. 동작 인식이 아닌 이미지 인식의 경우, 다양한 방법이 있습니다.

저해상도에서의 객체 및 얼굴 인식을 포함한 이미지 품질에 대한 연구[19]-[22], 흐림, 노이즈, 대비 변화, JPEG 압축 등 다양한 이미지 품질 저하에 대한 인식 모델의 영향에 대한 조사[23], [24]가 진행되었습니다. 동작 인식을 위해 저해상도 비디오에 대한 여러 가지 방법이 제안되었습니다[25], [26]; 그러나 이러한 접근 방식은 비디오 품질 저하의 영향을 무시하기 위해 이미지 향상을 통해 품질 저하 문제를 조절합니다.

따라서 본 연구에서는 모델 성능 평가에 일반적으로 사용되는 액션 데이터셋인 Kinetics400 [5]을 대상으로 동영상 화질과 액션 인식 모델의 성능 간의 트레이드 오프를 정량적으로 평가합니다. 하나는 검증 세트의 품질이 저하된 동영상에 대해 사전 학습된 모델을 검증하는 것이고, 다른 하나는 훈련 세트의 품질이 저하된 동영상에 대해 모델을 훈련하는 것입니다. 우리는 주로 정지 이미지 인코더인 JPEG와 동영상 인코더인 H.264/AVC를 통한 트랜스코딩과 관련된 품질 저하에 중점을 둡니다. 이러한 코덱이 더 높은 압축률로 이미지와 동영상을 트랜스코딩할 때 더 많은 아티팩트가 나타납니다.

- JPEG 트랜스코딩의 경우, 각 입력 비디오 프레임을 번호가 매겨진 JPEG 이미지 시퀀스로 저장합니다(프레임 속도가 30fps로 고정됨). 양자화 매개변수는 압축률을 변경하기 위해 지정됩니다. 사전 학습된 모델을 검증하는 실험을 용이하게 하기 위해 데이터 증강[27]을 사용하여 JPEG 압축을 시뮬레이션합니다. 즉, 원본 비디오 파일을 읽고 각 프레임에 JPEG 압축을 적용하여 품질 파라미터를 광범위하게 변화시킵니다. 실제로 모델 학습을 위한 실험에서는 제한된 범위의 품질 파라미터를 사용하여 비디오 프레임을 JPEG 파일로 트랜스코딩합니다.
- H.264/AVC 트랜스코딩의 경우, 입력 비디오를 감소된 비트 전송률로 다시 인코딩합니다. 동작 인식에 일반적으로 사용되는 키네틱스400과 같은 데이터 세트는 YouTube 기반이며, 다운로드한 동영상은 일반적으로 H.264/AVC로 인코딩되기 때문에 트랜스코딩에 H.264/AVC 코덱을 사용합니다.

이 두 가지 트랜스코딩 설정에 대한 자세한 내용은 섹션 III에서 설명합니다. 섹션 IV에서는 JPEG 및 H.264/AVC로 트랜스코딩된 검증 비디오에서 잘 알려진 사전 학습된 모델의 성능을 평가합니다. 그런 다음 섹션 V에서는 트랜스코딩이 트랜스코딩된 트레이닝 비디오에서 학습된 모델의 성능에 어떤 영향을 미치는지 보여줍니다.

II. 관련 작업

저해상도 이미지 인식. 자연 상태에서 촬영한 이미지와 원거리에서 촬영한 이미지와 같은 조건에서는 인식할 영역의 해상도가 상당히 작습니다. 간단한 접근 방식은 초해상도를 사용하여 저해상도 이미지를 고해상도 이미지로 변환한 다음 전제 인식 방법을 적용하는 것입니다 [22], [28]. 클래스별 도메인 지식을 사용하여 저해상도에 대처하고[29] 동시에 고품질 이미지를 사용하는 다른 방법도 제안되었습니다[30].

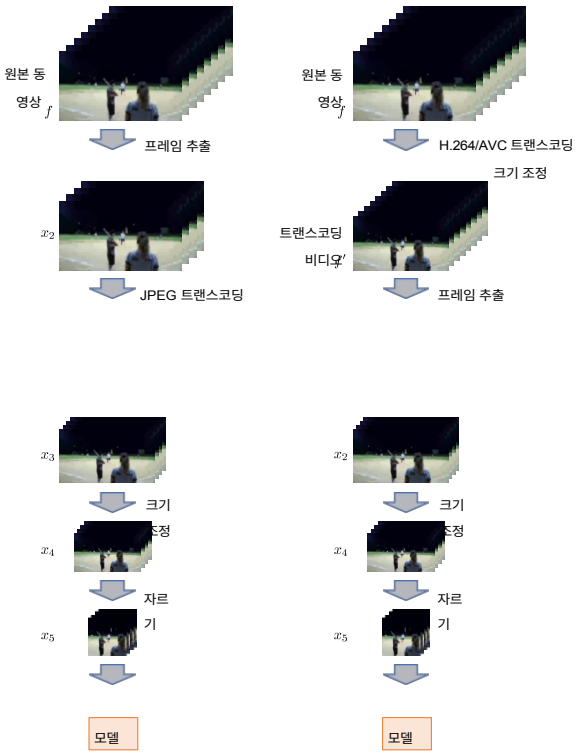


그림 2: (왼쪽) JPEG 트랜스코딩과 (오른쪽) H.264/AVC 트랜스코딩의 절차 개요.

이미지 품질 저하 분석. 이미지 품질이 저하되면 사전 학습된 CNN 모델을 사용하는 애플리케이션에서 인식 성능이 크게 저하될 수 있습니다. 일부 연구에서는 이미지의 다양한 변형(노이즈, 흐림, 압축 등)에 대해 일반 CNN 모델이 얼마나 잘 수행되는지 정량적으로 평가했습니다[24], [31]-[33]. 정량적 평가를 위해 이미지넷에 다양한 손상과 섭동을 적용한 데이터셋인 ImageNet-C/P[23]도 제안되었습니다[34]. 이 문제를 해결하기 위해 사전 심층 열화[35], 열화된 이미지로부터 재구조화된 이미지를 학습한 후 미세 조정[36] 등 다양한 방법이 제안되었습니다.

저해상도 비디오의 동작 인식. 저해상도 영상을 위한 초해상도는 저해상도 영상을 인식하기 위한 솔루션입니다 [25], [26]. 또한 일반적인 프레임 속도인 30fps보다 느린 저프레임 속도[37]를 처리하는 방법도 있습니다. 그러나 현재까지 저화질 영상의 성능 저하를 해결하기 위한 연구는 비디오 향상 네트워크 [38]를 제외하고는 많지 않습니다. 비디오 향상

알고리즘 1 JPEG 트랜스코딩 절차

1) 데이터 로더를 사용하여 입력 비디오 파일 f에서 비디오 클립 $x_1 \in \mathbb{R}^{sT_{\text{in}} \times 3 \times H_{\text{in}} \times W_{\text{in}}}$을 추출합니다. 여기서 T_{in} 는 클립의 프레임 수, s 는 프레임 사이의 보폭, H_{in} 와 W_{in} 는 비디오 프레임의 높이와 너비입니다. 프레임은 보폭이 s인 x_1 에서 균등하게 샘플링되어 클립 $x_2 \in \mathbb{R}^{T_{\text{in}} \times 3 \times H_{\text{in}} \times W_{\text{in}}}$이 됩니다.
2) 각 프레임에 대해 JPEG 압축 버전 x_3 생성 x_2 에서 지정된 압축 강도 cs로 압축합니다.
3) 가로 세로 비율을 유지하면서 크기가 조정된 클립의 짧은 면 x_4 이 지정된 크기 ss가 되도록 프레임 크기 x_3 를 조정합니다. x_4 의 크기는 다음과 같습니다;

$$x_4 \in \mathbb{R}^{T_{\text{in}} \times 3 \times ss \times \frac{H_{\text{in}}}{W_{\text{in}}} \times ss} \quad \begin{matrix} H_{\text{in}} \leq W_{\text{in}} \\ H_{\text{in}} \geq W_{\text{in}} \end{matrix} \qquad (1)$$

.

네트워크는 입력 프레임과 출력 프레임 사이의 픽셀 수준 손실과 지각 및 적대적 손실을 이용하여 압축된 저품질 비디오 프레임을 고품질 비디오 프레임으로 변환합니다. 이 연구와 달리 본 연구는 저화질 비디오에 적합하지 않은 동작 인식 모델을 실험적으로 분석하여 성능 트레이드 오프를 보여줍니다 .

III. 트랜스코딩 방법

이 섹션에서는 JPEG 및 H.264/AVC로 동영상을 트랜스코딩하는 방법에 대해 설명합니다. 트랜스코딩 절차의 개념은 그림 2에 나와 있습니다.

즉, x_4 는 x_3 의 원래 크기에 관계없이 ss 로 크기가 조정됩니다.

4) 프레임 중앙의 $H \times W$ 픽셀 영역을 자릅니다 x_4 . 이렇게 하면 클립 $x_5 \in \mathrm{R}^{T\ in \times 3 \times H \times W}$가 됩니다. 이 실험에서는 $H = W = 160$ 또는 $H = W = 224$ 입니다.

A. JPEG 트랜스코딩

먼저 각 입력 비디오 프레임의 JPEG 트랜스코딩을 시뮬레이션하는 방법을 설명합니다. 본 연구의 실험을 효율적으로 수행하기 위해, 비디오 파일의 프레임을 일련의 JPEG 이미지 파일로 저장하는 대신 비디오의 각 프레임을 읽고 JPEG 압축을 적용합니다. 이를 위해 압축 강도를 0에서 100 사이의 파라미터로 설정한 데이터 증강으로 `imgaug`[27]의 JPEG 압축을 적용합니다. 이 매개변수는 품질 계수의 역수이며, 100은 가장 높은 압축을 의미합니다.

그림 3은 압축 강도가 0에서 100 사이일 때 압축된 프레임의 모습을 보여줍니다. 0에서 80까지는 눈에 띄는 아티팩트가 없지만, 압축 강도가 80을 초과하면 블록 노이즈가 뚜렷하게 나타나고 화질이 크게 저하되는 것을 확인할 수 있습니다.

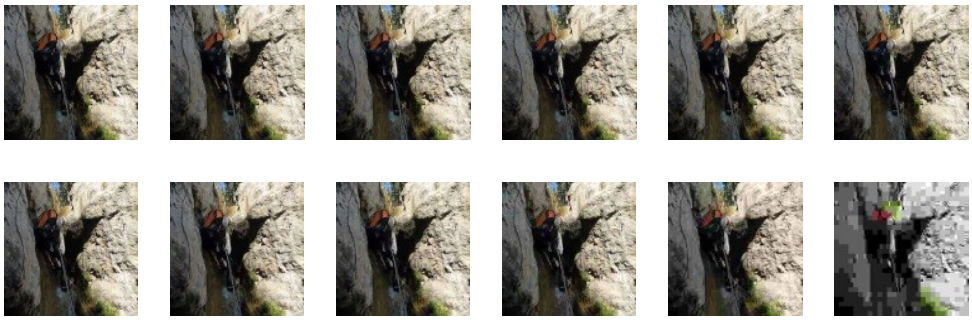
알고리즘 1은 JPEG 압축, 크기 조정 및 비디오 프레임 자르기를 적용하는 데 사용되는 절차를 보여줍니다.

B. H.264/AVC 트랜스코딩

다음으로 H.264/AVC 코덱을 사용하여 입력 비디오 파일을 트랜스코딩하는 방법을 설명합니다. 본 연구에서는 두 가지 품질 파라미터, 즉 그룹 오브 픽처(GOP) 크기와 일정한 비율 계수(CRF)를 변경하여 트랜스코딩을 수행합니다. 실험용 비디오는 `FFmpeg`[39]를 사용하여 x264로 트랜스코딩합니다.[1] 로 트랜스코딩되며, 이는 잘 확립된 H.264/ACV 코덱입니다.
-사전 설정 매체[2] 를 사용하세요.

GOP 크기가 클수록 프레임 간 간격(I-프레임)이 길어집니다. GOP 크기가 1이면 모든 프레임이

[1] https://www.videolan.org/developers/x264.html
2https://trac.ffmpeg.org/wiki/Encode/H.264



(A) 원본 (B) 0 (C) 10 (D) 20 (E) 30 (F) 40

(G) 50(H) 60 (I) 70 (J) 80 (K) 90 (L) 100

그림 3: 다양한 압축 강도 값(숫자가 높을수록 프레임이 더 많이 압축됨)의 JPEG 트랜스코딩에 따른 이미지 품질 변화.

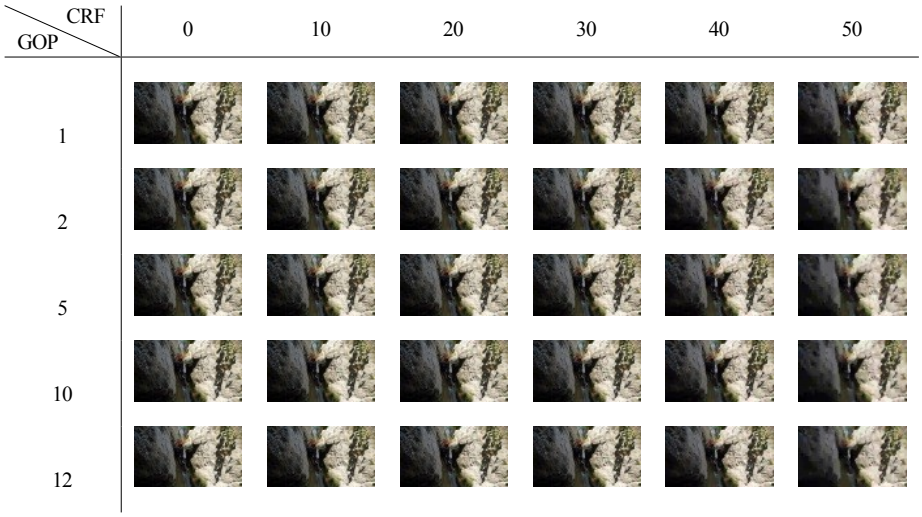


그림 4: GOP 크기와 CRF 값이 다른 H.264/AVC 트랜스코딩으로 인한 화질 변화.

표 1: 각 모델에 대한 입력 비디오 클립 생성을 위한 매개변수.

	ss	*W*, *H*	Tin	*s*
X3D-XS	181	160	4	12
X3D-S	181	160	13	6
X3D-M	256	224	16	5
슬로우패스트 R50	256	224	32	2
슬로우패스트 R100	256	224	32	2
3D ResNet R50	256	224	8	8
타임포머	256	224	8	32
비디오 스윈-B/T	256	224	32	2

은 I-프레임으로 인코딩되며 동영상 파일 크기가 커집니다. CRF 값은 비디오 품질을 제어하며, CRF의 유효 범위는 0~51(정수)로, 0은 비압축, 51은 최대 압축이며 기본값은 23입니다. 그림 4는 GOP 크기와 CRF를 변경하여 트랜스코딩된 비디오 프레임의 화질 변화를 보여줍니다. CRF 40까지의 모든 GOP 크기에서는 시각적으로 큰 변화가 관찰되지 않지만 CRF 50에서는 눈에 띄는 품질 저하가 관찰됩니다.

알고리즘 2는 비디오 프레임 크기 조정 및 트랜스코딩에 사용되는 단계를 보여줍니다.

IV. 사전 학습된 모델 평가

이 섹션에서는 검증 세트의 품질이 저하된 동영상을 사용하여 동작 인식을 위해 사전 학습된 일반적인 모델의 성능 평가를 분석합니다.

A. 실험적 설정

데이터 세트. 22,000개의 동영상으로 구성된 학습 세트, 18,000개의 동영상으로 구성된 검증 세트, 35,000개의 동영상으로 구성된 테스트 세트로 구성된 가장 일반적인 동작 인식 데이터 세트인 Kinetics400[5]을 사용했으며, 400가지 범주의 인간 동작이 포함되어 있습니다. 각 동영상은 YouTube에서 추출했으며, 동영상에서 액션 부분은 10초 분량으로 잘라냈습니다. 이 연구에서는 검증 세트에 19880개의 동영상을 사용했습니다.[3] Kinetics-700 챌린지 2021에 배포된 19880개의 동영상[4].

모델. 조명 분야에서 일반적으로 비교에 사용되는 다음과 같은 최신 CNN 모델을 사용했습니다: X3D-M/S/XS [14], SlowFast(R50/R101) [15], 3D ResNet(R50) [16], 비전 트랜스포머(ViT) [40] 기반의 최신 모델, TimeSformer [17] 및 비디오

[3] https://github.com/cvdfoundation/kinetics-dataset
4https://eval.ai/web/challenges/challenge-page/1054/overview

알고리즘 2 H.264/AVC 트랜스코딩 절차

- (준비) 입력 비디오 파일 f를 H.264/AVC로 트랜스코딩하여 지정된 GOP 크기와 CRF를 가진 비디오 파일 f 을 생성합니다. 동시에 알고리즘 1의 3단계와 유사한 방식으로 프레임 크기를 조정하여 ss = 360픽셀(360p)로 설정합니다;

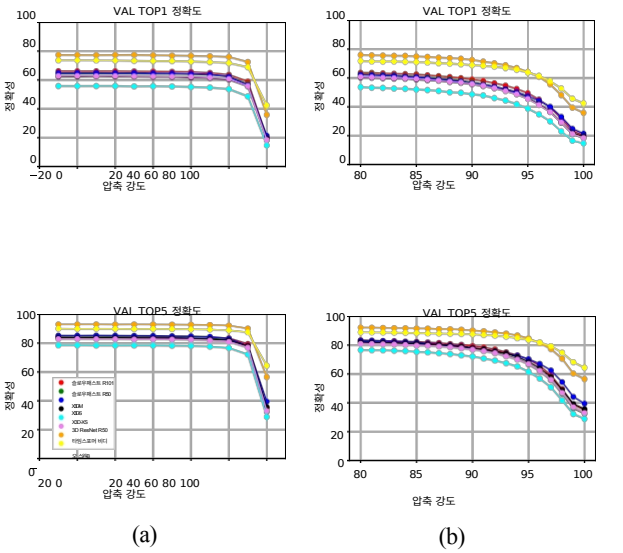
$$f \in \begin{cases} \mathbb{R}^{T_{\text{in}} \times 3 \times ss \times ss \frac{\text{Win}}{\text{Hin}}} & \text{힌} \leq \text{승리} \\ \mathbb{R}^{\text{주석} \times 3 \times ss \frac{\text{힌}}{\text{in}} \times ss} & \text{Hin} \geq W_{\text{in}} \\ \mathbb{R}^{\text{주석} \times 3 \times \text{힌} \times \text{윈}} & \min(H_{\text{in}}, W_{\text{in}}) < ss, \end{cases} \quad (2)$$

여기서 T_{in} 는 f의 프레임 수 , H_{in} 와 W_{in} 는 f의 프레임의 높이와 너비입니다. 다시 말해 f 의 원래 크기가 SS보다 큰 경우에만 SS로 크기가 조정됩니다.

1) 트랜스코딩된 동영상 파일에서$_2$ 동영상 클립을 추출합니다.
f 알고리즘 1의 1단계에서와 같이 데이터 로더를 사용합니다.
2) 알고리즘 1의 3단계에서와 같이 프레임 크기를 조정합니다.
3) 알고리즘 1의 4단계에서와 같이 중앙 영역을 자릅니다.

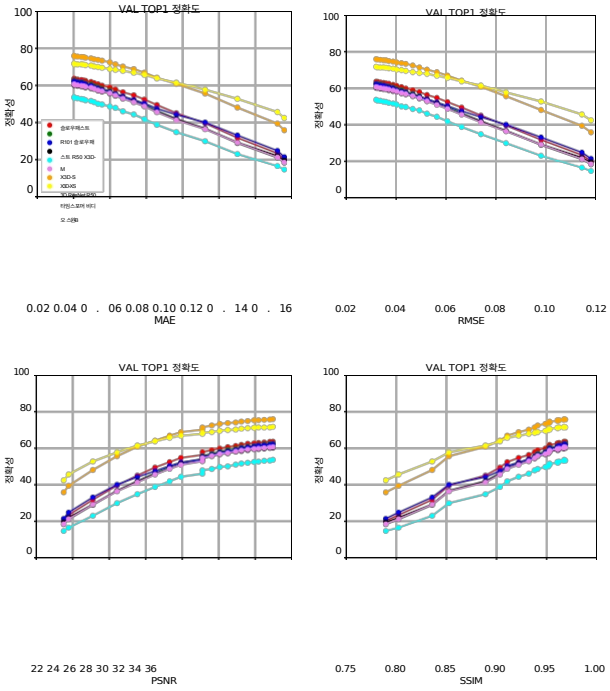
스윈 트랜스포머(비디오 스윈-B) [18]. 이 모델들은 Kinetics400의 훈련 세트의 원본 비디오에 대해 사전 훈련되었습니다. 사전 훈련된 X3D, SlowFast 및 3D ResNet 모델은 PyTorchVideo [41] 모델 동물원에서 얻었으며, TimeSformer 및 Video Swin 모델은 다음과 같습니다.
공식 리포지토리에서 가져옵니다.[5] [6] 표 1은
매개변수를 사용하여 각 모델에 대한 입력 비디오 클립을 생성합니다. 일반적으로 클립의 시작 프레임은 무작위로 선택되며 여러 클립으로 테스트 시간을 늘려 모델을 평가하지만(이를 멀티뷰 테스트라고 함), 이 실험에서는 시작 프레임을 비디오의 시작 부분으로 고정하여 단일 뷰로 평가를 재현할 수 있도록 했습니다.

메트릭. 상위 1~5위를 다음과 같이 분류하여 보고합니다.
는 동작 인식을 위한 일반적인 지표입니다[14]-[18].



(a) (b)

그림 5: JPEG 트랜스코딩으로 성능이 저하된 비디오에 대한 각 모델의 상위 1(위) 및 상위 5(아래) 성능. (a) 압축 강도 cs 가0에서 100으로 10 증가했습니다.
(b)는 80에서 100으로 1 증가했습니다. cs = -10 는 원본 동영상을 나타냅니다.



B. JPEG 트랜스코딩 결과

그림 5(a)는 JPEG 압축 강도 cs를 0에서 100으로 10씩 증가시켰을 때의 성능을 보여줍니다. 모든 모델에서 성능 저하 추세가 비슷하다는 것을 알 수 있습니다. $cs = 70$까지는 성능에 큰 변화가 없으나 80 이후부터 점차 성능이 저하되어 $cs = 100$에서는 원본 영상의 절반 수준으로 인식률이 감소합니다. 그림 5(b)는 80에서 100까지의 미세한 간격에 따른 성능을 나타낸 것으로, 100에 가까워질수록 성능이 점차 감소하는 것을 확인할 수 있습니다. 그림 3에서와 같이

[5] 사전 학습된 모델은 다음 저장소에서 제공됩니다.
https://pytorch.org/hub/facebookresearch_pytorchvideo_resnet/
https://pytorch.org/hub/facebookresearch_pytorchvideo_slowfast/
https://pytorch.org/hub/facebookresearch_pytorchvideo_x3d/
https://github.com/facebookresearch/TimeSformer
https://github.com/SwinTransformer/Video-Swin-Transformer

[6] 상용 플랫폼에서 API로 제공되는 상용 모델은 사용자에게 블랙박스로 제공되며, 사용자도 모르는 사이에 기반 모델 및 알고리즘이 변경될 수 있으므로 제외합니다.

그림 6: 그림 6: JPEG 트랜스코딩의 최고 성능과 품질(MAE, RMSE, PSNR, SSIM)의 관계(cs 80~100). MAE와 RMSE의 경우 값이 작을수록 품질이 우수하고, PSNR과 SSIM의 경우 값이 클수록 품질이 좋습니다. 품질은 디코딩, 크기 조정 및 JPEG 트랜스코딩된 비디오 프레임을 디코딩 및 크기 조정된 원본 비디오 프레임으로 참조하여 계산했습니다.

으로 트랜스코딩한 이미지와 $cs = 80$ 및 $cs = 90$으로 트랜스코딩한 이미지의 시각적 변화는 크지 않습니다. 그러나 모델의 성능은 $cs = 80$부터 저하되며, 원본 동영상에 비해 $cs = 90$에서는 성능이 약 10% 떨어집니다.

다음으로 성능과 트랜스코딩 결과의 품질 간의 관계를 조사합니다. 그림 6은 평균 절대 오차(MAE), 평균 제곱근 오차(RMSE), 피크 신호 잡음비(PSNR) 및 구조 유사성(SSIM)에 대한 성능을 보여줍니다. 결과는 품질과 성능 간의 선형적인 관계를 보여줍니다. 즉, 모델의 성능은 품질에 비례합니다.

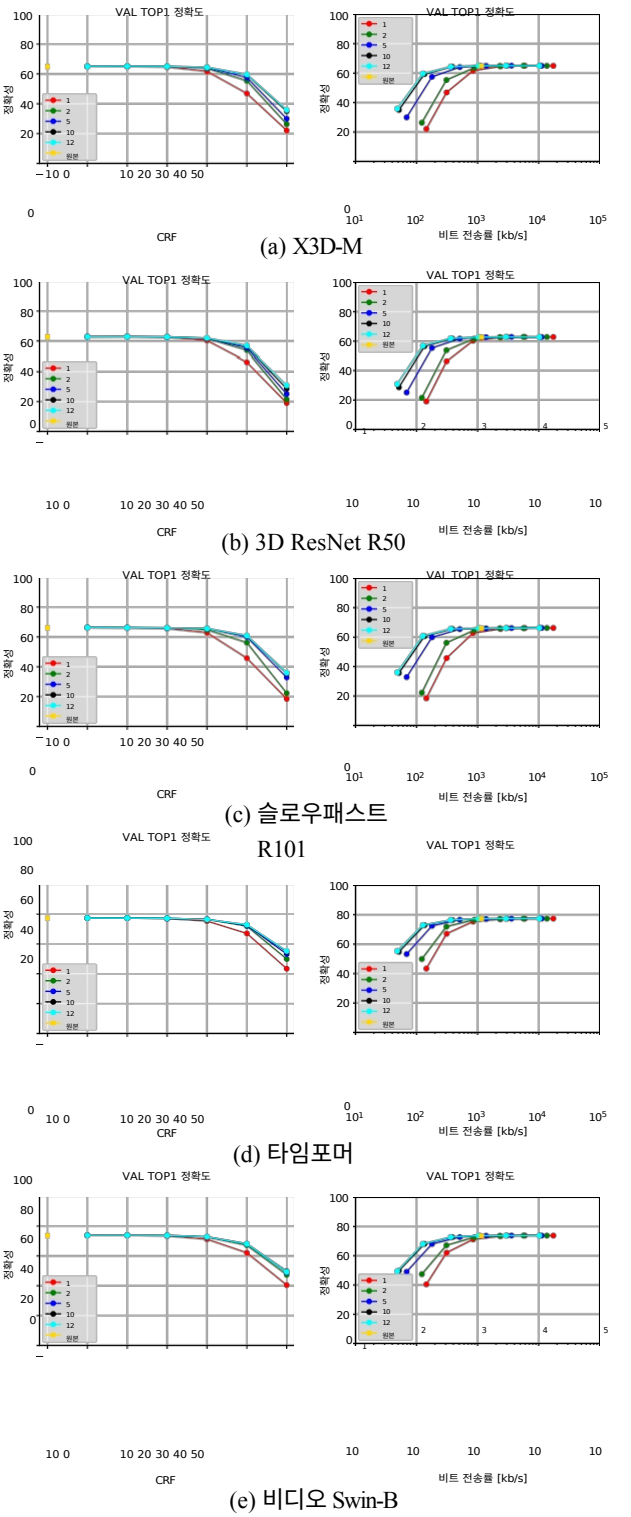
를 입력합니다. MAE, RMSE, SSIM의 경우 X3D-M의 최상위 1 성능의 선형 근사치의 *기울기* a와 *절편* b는 각각 (*a, b*) = (-338, 76.4), (-460, 77.6), (221, -152.0)입니다. 성능이 각각 338/256 = 1.32%씩 감소한다고 말할 수 있습니다. 1/256 ≈ 0.004의 MAE 증가, 1.80%의 RMSE 증가 (입력 프레임의 픽셀 값 범위가 [0, 255]가 아닌 [0, 1]이므로 256으로 나눕니다). 또한 SSIM이 0.01 감소할 때마다 성능이 2.21%씩 저하되는 것을 볼 수 있습니다.

예상대로 ViT 기반 모델이 CNN 기반 모델보다 더 나은 성능을 보였습니다. 하지만 흥미롭게도 비디오 스윈-B 당 최악의 품질에서 타임포머보다 더 나은 폼을 형성합니다(*cs* > 95). 그림 6에서 비디오 스윈-B의 기울기가 다른 모델보다 작다는 것을 알 수 있습니다. 이는 Video Swin-B와 같이 모델 아키텍처를 신중하게 설계하지 않으면 ViT 기반 모델의 성능이 CNN 기반 모델만큼 저하될 수 있음을 나타냅니다.

C. H.264/AVC 트랜스코딩 결과

그림 7은 검증 세트의 H.264/AVC로 트랜스코딩된 비디오에 대한 다양한 모델의 성능을 보여줍니다. 모든 모델에서 CRF가 20 미만일 때는 성능 차이가 거의 없습니다. CRF가 클수록 GOP 크기가 클수록 성능이 더 느리게 감소하는 것을 볼 수 있습니다.
GOP의 경우 최악의 성능이 관찰됩니다.
크기를 1로 설정하면 비디오에 I-프레임만 있습니다. SlowFast-R101의 경우, GOP 크기가 2인 경우 CRF 40에서 50으로 넘어가면서 X3D-M 및 Slow-R50에 비해 성능 저하가 심하게 나타납니다. 트랜스포머 기반 모델의 경우 GOP 크기 차이(2보다 큰 경우)로 인한 성능 저하는 CNN-에 비해 크지 않습니다.
기반 모델입니다.

표 2는 성능 저하가 어떻게 비교되는지 보여줍니다. 을 원본 비디오의 성능과 비교합니다. GOP 크기가 5보다 큰 경우, CRF 30에서는 성능 저하가 0.5%~1.0% 정도이지만, CRF 40에서는 모든 모델에서 5% 이상으로 커집니다. CRF 50의 경우 CNN 모델의 경우 원본 성능의 50%, ViT 기반 모델의 경우 70%까지 성능이 떨어집니다.



(a) X3D-M

(b) 3D ResNet R50

(c) 슬로우패스트 R101

(d) 타임포머

(e) 비디오 Swin-B

그림 7의 결과는 모든 비디오에 대한 평균 성능을 보여주며, 모든 액션 범주에 대해 동일한 방식으로 성능이 저하되는지 여부는 명확하지 않습니다. 그림 8은 CRF 값이 다른 각 클래스에 대한 X3D-M의 상위 1% 성능을 보여줍니다(GOP 크기는 12로 고정됨). 카테고리별로 인식률이 크게 다르므로 이

그림은 원본 비디오의 기준(100으로 설정)에 대한 상대적인 성능을 보여줍니다(CRF -10에서). 각 CRF에 대해 서로 다른 카테고리에 해당하는 400개의 원이 그려져 있습니다. 이 그림에서 볼 수 있듯이 CRF가 30 미만인 경우 모든 카테고리에서 성능 저하가 크게 나타나지 않습니다. 일부 카테고리는 CRF 40에서 성능이 크게 감소하고(흥미롭게도 일부 카테고리는 성능이 증가함), 거의 모든 카테고리는 CRF 50에서 큰 성능 저하를 보입니다.

그림 7: H.264/AVC로 트랜스코딩된 비디오의 상위 1순위 성능. 왼쪽에는 다양한 CRF 값에 대한 모델의 성능이 표시되어 있고(CRF -10의 값은 원본 비디오를 나타냄), 오른쪽에는 비트 전송률에 대한 성능이 표시되어 있습니다.

H.264/AVC로 트랜스코딩된 동영상의 총 파일 크기는 표 3에 나와 있습니다. GOP 크기가 1이고 CRF가 0(무손실)인 경우 파일 크기(약 400GB)가 원본 동영상(약 30GB)보다 13배 커집니다. 파일 크기 감소 측면에서 동영상 트랜스코딩의 이점을 누리려면 CRF가 30 이상이어야 한다는 것을 알 수 있습니다. GOP 크기가 12인 CRF 30의 경우 파일 크기가 원본 영상의 약 40%로 줄어들어 약 0.5%의 성능 손실이 발생했습니다. 20보다 작은 CRF로 트랜스코딩하면 성능에는 변화가 없지만 파일 크기가 원본 동영상과 같거나 더 커져 이점이 없습니다.

표 2: H.264/AVC로 트랜스코딩된 동영상의 상위 1% 성능. 괄호 안의 값은 원본 비디오의 참조 성능(모델명 측면에 표시됨)과의 차이를 나타냅니다. 음영 처리된 셀은 1.0% 이상의 성능 저하를 나타냅니다.

GOP \ CRF	0	10	20	30	40	50
1	64.99 (+0.07)	65.03 (+0.11)	64.60 (-0.32)	61.70 (-3.22)	46.89 (-18.03)	22.11 (-42.81)
2	64.99 (+0.07)	65.06 (+0.14)	64.93 (-0.01)	63.68 (-1.24)	55.35 (-9.57)	26.35 (-38.57)
5	64.99 (+0.07)	65.08 (+0.16)	64.97 (+0.05)	64.10 (-0.82)	57.53 (-7.57)	29.95 (-34.97)
10	64.99 (+0.07)	65.04 (+0.12)	64.98 (+0.06)	64.41 (-0.51)	59.37 (-5.55)	34.98 (-29.94)
12	64.99 (+0.07)	65.10 (+0.18)	65.03 (+0.11)	64.39 (-0.53)	59.70 (-5.22)	35.88 (-29.04)

(a) X3D-M (64.92%)

GOP \ CRF	0	10	20	30	40	50
1	62.79 (+0.05)	62.80 (+0.06)	62.43 (-0.31)	60.31 (-2.43)	46.27 (-16.47)	19.01 (-43.73)
2	62.80 (+0.06)	62.88 (+0.14)	62.74 (±0.00)	61.68 (-1.06)	53.86 (-8.88)	21.49 (-41.25)
5	62.79 (+0.05)	62.87 (+0.13)	62.66 (-0.08)	61.72 (-1.02)	55.39 (-7.35)	25.10 (-37.64)
10	62.79 (+0.05)	62.87 (+0.13)	62.74 (±0.00)	61.75 (-0.99)	56.51 (-6.23)	28.64 (-34.1)
12	62.79 (+0.05)	62.86 (+0.12)	62.70 (-0.04)	61.95 (-0.79)	56.96 (-5.78)	30.87 (-31.87)

(b) 3D ResNet R50 (62.74%)

GOP \ CRF	0	10	20	30	40	50
1	66.26 (+0.11)	66.17 (+0.02)	65.57 (-0.58)	62.80 (-3.35)	45.75 (-20.4)	18.45 (-47.7)
2	66.26 (+0.11)	66.12 (+0.03)	66.02 (-0.13)	64.74 (-1.41)	56.16 (-9.99)	22.26 (-43.89)
5	66.26 (+0.11)	66.18 (+0.3)	65.93 (-0.22)	65.47 (-0.68)	59.91 (-6.24)	32.82 (-33.33)
10	66.26 (+0.11)	66.20 (+0.05)	66.06 (-0.09)	65.60 (-0.55)	60.62 (-5.53)	35.66 (-30.49)
12	66.26 (+0.11)	66.15 (±0.00)	65.99 (-0.16)	65.65 (-0.50)	60.89 (-5.26)	36.04 (-30.11)

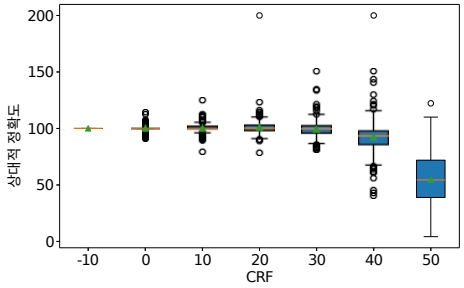
(c) SlowFast R101 (66.15%)

GOP \ CRF	0	10	20	30	40	50
1	77.37 (+0.02)	77.32 (-0.03)	76.93 (-0.42)	75.33 (-2.02)	67.05 (-10.3)	43.41 (-33.9)
2	77.37 (+0.02)	77.34 (-0.01)	77.15 (-0.20)	76.53 (-0.82)	71.92 (-5.43)	49.89 (-27.5)
5	77.37 (+0.02)	77.33 (-0.02)	77.06 (-0.29)	76.56 (-0.79)	72.41 (-4.94)	53.25 (-24.1)
10	77.37 (+0.02)	77.30 (-0.05)	77.25 (-0.10)	76.37 (-0.98)	72.85 (-4.50)	54.81 (-22.5)
12	77.37 (+0.02)	77.34 (-0.01)	77.22 (-0.12)	76.46 (-0.89)	72.95 (-4.46)	55.33 (-22.0)

(d) TimeSformer (77.35%)

GOP \ CRF	0	10	20	30	40	50
1	73.79 (+0.17)	73.82 (+0.20)	73.36 (-0.26)	71.19 (-2.43)	62.09 (-11.5)	40.45 (-33.2)
2	73.79 (+0.17)	73.81 (+0.19)	73.71 (+0.09)	72.63 (-0.99)	67.15 (-6.47)	47.40 (-26.2)
5	73.79 (+0.02)	73.83 (+0.21)	73.71 (+0.09)	72.74 (-0.88)	68.04 (-5.58)	49.05 (-24.6)
10	73.79 (+0.02)	73.82 (+0.20)	73.69 (+0.07)	72.76 (-0.86)	68.09 (-5.53)	49.76 (-23.9)
12	73.79 (+0.02)	73.84 (+0.22)	73.76 (+0.14)	72.82 (-0.80)	68.11 (-5.51)	49.37 (-24.3)

(e) 비디오 스윈-B (73.62%)



Kinetics-700 챌린지 2021의 트레이닝 세트에서 240194개의 비디오로 모델을 학습시키고, 검증 세트에서 다음과 같은 검증 실험을 통해 모델을 평가했습니다.

V. 행동 인식 모델 학습

이전 섹션(4절)에서는 *원본 동영상에* 대해 사전 학습된 모델을 사용하여 JPEG 또는 H.264/AVC로 트랜스코딩된 검증 세트의 동영상 성능을 평가한 결과를 보여드렸습니다. 여기에서는 훈련 세트의 *트랜스코딩된 비디오에* 대해 훈련된 모델의 성능을 보여줍니다.

그림 8. CRF 값이 다른 각 클래스의 상대적 상위 1위 성능. 원본 비디오의 성능(CRF -10으로 표시됨)을 기준(100)으로 표시하고 400개 클래스에 대한 상대적 성능을 원으로 표시했습니다. 모델은 X3D-M이고 GOP 크기는 12로 고정되었습니다.

표 3: 트랜스코딩된 19880개의 Kinetics400 유효성 검사 비디오의 총 파일 크기(GB). 프레임은 짧은 쪽에서 최대 360 픽셀로 크기가 조정되었습니다. 음영 처리된 셀은 원본 동영상의 총 파일 크기인 30.56GB보다 파일 크기가 증가했음을 나타냅니다.

GOP \ CRF	0	10	20	30	40	50
1	417.91	144.22	59.42	23.02	10.30	6.33
2	327.40	137.67	58.12	23.97	10.26	5.81
5	269.13	88.31	35.67	15.08	7.20	4.56
10	249.66	74.47	28.80	12.08	6.13	4.14
12	247.20	72.05	27.59	11.54	5.93	4.06

사전 학습된 모델.

아래에 사용된 모델은 X3D-M [14], 3D ResNet R50 [16], SlowFast R101 [15], 타임스포머 [17], 비디오 스윈 트랜스포머(비디오 스윈-T) [18]입니다. [7]로 사전 훈련된 Kinetics400[8]. Adam [42]은 $\beta_1 = 0.9$, $\beta_2 = 0.999$, 학습률 10^{-4} 의 CNN 기반 모델에 사용되었으며, SGD는 모멘텀 0.9, 학습률 5 - 10^{-4} 의 Transformer 기반 모델에 사용되었습니다. 두 경우 모두 5 - 10^{-5} 의 가중치 감쇠를 사용했습니다. 표 1에 표시된 파라미터는 훈련용 클립의 시작 프레임을 무작위로 선택하여 각 모델에 대한 입력 비디오 클립을 생성하는 데 사용되었습니다. 모델에 대한 입력 비디오 클립의 프레임 샘플링은 화면비를 유지하면서 짧은 면의 크기를 [256, 320] 픽셀 범위에서 임의로 선택한 크기로 조정한 다음 224 × 224 픽셀 패치를 임의로 자르고 50% 비율로 수평으로 뒤집는 것을 제외하고는 III-B 섹션의 단계와 같이 수행했습니다. 배치 크기는 8개, 훈련 에포크는 5개였습니다.

유효성 검사를 위해 섹션 IV-A에서와 동일한 절차를 사용하여 유효성 검사 클립을 생성했습니다.

표 4: 240194개의 열차 동영상과 19880개의 검증 동영상에서 트랜스코딩된 JPEG 파일의 총 파일 크기(GB). 프레임은 짧은 쪽에서 최대 360픽셀로 크기가 조정되었습니다.

-q	-1	10	17	40
cs	N/A	70	80	90
기차 파일 크기	3504	937	726	526
VAL 파일 크기	293	77	60	43

표 5: JPEG 트랜스코딩으로 저하된 Kinetics400 비디오에 대해 학습 및 평가된 모델의 상위 1% 성능.

-q	-1	10	17	40
cs	N/A	70	80	90
X3D-M	59.09	58.90	59.12	57.59
3D ResNet R50	52.40	50.89	49.90	49.15
슬로우패스트 R101	52.42	54.96	53.59	53.35
타임포머	76.60	76.05	75.55	74.20
비디오 Swin-T	68.43	67.45	66.93	65.33

A. JPEG 트랜스코딩 결과

이 실험에서는 이전 섹션의 평가 실험에서와 같이 데이터 증강을 적용하는 대신 원본 비디오에서 트랜스코딩된 번호가 매겨진 JPEG 이미지 시퀀스를 사용했습니다. 트랜스코딩의 품질 계수는 고정 프레임 속도 30fps의 `FFmpeg`[39]를 사용하여 지정했습니다. 이전 섹션에서 압축 강도 cs가 80일 때 성능이 저하되는 것을 확인했기 때문에 70, 80, 90의 세 가지 값을 선택했습니다. cs 값과 `FFmpeg` 옵션의 품질 스케일 `-q`를 맞추기 위해 예비 보정을 통해 `-q` = 10, 17, 40을 사용했습니다(`-q`의 값은 -1~65 범위이며, 값이 작을수록 품질이 좋아짐). 표 4는 해당 cs 값과 함께 트랜스코딩된 JPEG 파일의 총 크기를 보여줍니다. 참고로, 최대 JPEG 품질(즉, 설정된
-q에서 -1)도 표에 표시되어 있으며, 이는 데이터 집합 준비의 일반적인 방법입니다.

표 5는 다양한 cs 설정에 따른 성능을 보여줍니다. X3D-M의 경우 cs = 70에서의 성능은 최고 품질의 JPEG 트랜스코딩 성능과 거의 동일하며 그 차이는 0.1%에 불과합니다. cs가 증가함에 따라 성능이 감소하여 cs = 90에서 3%의 성능 저하가 발생했습니다. 이 결과는 사전 학습된 모델을 사용한 검증 실험의 결과와 일치했습니다. 3D ResNet과 슬로우패스트의 경우 cs = 70에서 성능이 약 1.5% 저하되어 X3D-M보다 약간 떨어지지만, 검증 실험 결과에서 크게 벗어나지 않습니다.

표 6: 트랜스코딩된 유효성 검사 세트의 상위 1퍼센트 성능과 Kinetics400의 트랜스코딩된 트레이닝 세트에서 학습된 모델. 두 세트 모두 동일한 CRF 설정으로 H.264/AVC로 트랜스코딩되었습니다.

CRF	원본	20	30	40
X3D-M	59.53	58.99	58.49	56.28
3D ResNet R50	51.29	52.63	50.86	47.05
슬로우패스트 R101	54.58	54.85	53.58	50.44
타임포머	76.90	77.06	76.58	74.80
비디오 Swin-T	68.54	68.94	67.97	66.01

표 7: 240194개의 열차 동영상과 H.264/ACV로 트랜스코딩된 Kinetics400의 검증 동영상 19880개의 총 파일 크기(GB). 프레임은 짧은 쪽에서 최대 360픽셀로 크기가 조정되었습니다. GOP 길이는 12로 고정되었습니다.

CRF	원본	20	30	40
기차 파일 크기	349	313	131	68
VAL 파일 크기	31	28	12	6

B. H.264/AVC 트랜스코딩 결과

H.264/AVC로 트랜스코딩된 학습용 비디오를 사용한 실험의 경우, 사전 학습된 모델에 대한 이전 실험과 동일한 방식으로 성능을 평가했습니다. 검증 세트에 포함된 19880개의 비디오가 트랜스코딩되었습니다.

[7] 효율적인 트레이닝을 위해 스윈-B 대신 더 작은 스윈-T를 사용했습니다.
[8]사전 학습과 다운스트림 작업 모두에 동일한 데이터 세트를 사용했기 때문에 이러한 결과는 일반적이지 않습니다. 그러나 사전 학습을 통해 얻은 성능보다 성능이 어떻게 감소하는지 궁금합니다.

를 훈련 세트의 해당 트랜스코딩 설정과 함께 사용합니다. 예를 들어, CRF 30으로 트랜스코딩된 검증 비디오는 CRF 30으로 트랜스코딩된 훈련 비디오로 학습된 모델을 평가하는 데 사용되었습니다.

앞선 섹션의 평가 실험 결과를 바탕으로 GOP 크기를 12로 고정하고 CRF 30을 표준 압축 설정으로 사용했으며, CRF 30을 설정하면 성능 저하가 상대적으로 적지만 파일 크기가 크게 줄어드는 합리적인 절충점이 있으므로 CRF를 20, 30, 40으로 변경해 보았습니다.

표 6은 CRF 설정이 다른 모델의 성능을 보여줍니다. CRF 30을 사용하면 원본 동영상에 비해 성능이 1% 정도 저하됩니다. 그러나 CRF 40의 성능은 3% 이상 감소하여 과도한 트랜스코딩이 인식률에 미치는 부정적인 영향을 확인할 수 있었습니다.

표 7은 훈련 세트에서 트랜스코딩된 영상의 총 파일 크기를 보여주는데, CRF 30으로 트랜스코딩하면 파일 크기가 약 40%로 줄어듭니다. 따라서 CRF 30은 성능 저하를 약 1%로 유지하면서 파일 크기를 크게 줄인다고 할 수 있습니다. 이는 사전 학습된 모델의 평가 실험에서 관찰된 결과와 일치합니다.

C. H.264/AVC를 사용한 HDDB51의 결과

Kinetics400에 대한 실험 외에도 다른 데이터 세트인 HMDB51을 사용하여 H.264/AVC 트랜스코딩을 사용한 학습에서도 유사한 추세가 관찰되는지 확인했습니다.

HMDB51[13]은 3.6k 개의 교육용 비디오와 1.5k 개의 검증용 비디오로 구성되어 있으며 51개의 인간 행동 범주로 분류되어 있습니다. 각 비디오는 영화, 웹, YouTube 등 다양한 출처에서 수집되었습니다. 가장 짧은 영상은 1초 미만, 가장 긴 영상은 약 35초이며, 대부분의 영상은 1초에서 5초 사이로 평균 길이는 3.15초입니다. 프레임의 짧은 쪽은 240픽셀로 크기를 조정했습니다. 이 실험에서는 일반적으로 보고되는 첫 번째 분할을 사용했습니다. 훈련 및 검증 절차는 Kinetics400에 대한 실험과 동일했습니다(섹션 V-B),

표 8: 트랜스코딩된 검증 세트의 상위 1% 성능과 트랜스코딩된 HMDB51의 트레이닝 세트에 대해 학습된 모델. 두 세트 모두 동일한 CRF 설정으로 H.264/AVC로 트랜스코딩되었습니다.

CRF	원본	20	30	40
X3D-M	69.24	70.22	68.00	64.99
3D ResNet R50	49.08	48.17	48.10	45.68
슬로우패스트 R101	59.22	64.20	60.54	58.97
타임포머	71.73	71.99	71.66	67.93
비디오 Swin-T	74.21	75.52	75.33	70.22

표 9: H.264/ACV로 트랜스코딩된 HMDB51의 훈련 및 검증 비디오의 총 파일 크기(MB). GOP 길이는 12로 고정되었습니다.

CRF	원본	20	30	40
기차 파일 크기	1153	852	265	101
파일 크기 테스트	472	346	108	42

각 모델의 마지막 선형 레이어가 미세 조정을 위해 무작위 초기화로 대체되고 합리적인 수렴을 위해 훈련 에포크를 50으로 설정했다는 점을 제외하고는 동일합니다.

표 8은 Kinetics400의 경우 표 6과 같이 CRF 설정이 다른 모델의 성능을 보여줍니다. 결과는 비슷한 경향을 보였는데, CRF 30으로 트랜스코딩할 경우 성능 손실이 크지 않았고(약 1%), CRF 40에서는 성능이 2%~4% 정도 감소했습니다.

표 9는 HMDB51의 트랜스코딩된 영상의 총 파일 크기를 보여주는데, CRF 30으로 트랜스코딩하면 파일 크기가 약 22%로 줄어들어 표 7에 표시된 Kinetics400의 30%보다 작아집니다.

VI. 토론 및 제한 사항

위의 실험은 비디오 파일로 수행했지만, 소개에서 언급했듯이 네트워크 연결 상태가 좋지 않은 상태에서 전송되는 비디오 스트림의 경우에도 비슷한 결과를 얻을 수 있습니다. 그림 7은 비트 전송률이 감소할 때 성능이 어떻게 저하되는지 보여줍니다. 원본 비디오 파일의 비트 전송률은 약 1000kb/s(해상도는 최대 360p)이며, 사전 학습된 모델은 이보다 10배 낮은 비트 전송률에서도 약간의 성능 저하가 있는 것으로 나타났습니다. 이 결과는 엣지 디바이스에서 전송되어 클라우드에서 처리되는 비디오 스트림에 대해 고품질 비디오 파일로 학습된 모델을 배포할 때 성능을 보장할 수 있는 열쇠가 될 수 있습니다.

파일 크기 측면에서 볼 때, 이 결과는 좋은 비용 대비 성능 절충안을 제시할 수 있습니다. 클라우드에서 학습을 수행할 때 대용량 비디오 데이터 세트를 클라우드 스토리지에 저장하는 것은 저렴하지 않을 수 있지만, 약간의 성능 저하를 감수하는 대신 크기를 줄일 수 있습니다. 현재 학술 연구에서 행동 인식 모델을 훈련하는 데 JPEG 파일을 사용하는 것이 일반적인 접근 방식일 수 있으며, 이는 물리적 서버에서 수행되는 경우가 많습니다. 이 경우, 표 4에 표시된 것처럼 최고 품질의 JPEG 파일의 총 크기는 3.5TB 이상으로, 느린 SATA3-SSD에는 간신히 들어갈 수 있지만 더 빠른 NVMe-SSD(현재 주류 크기는 2TB보다 작음)에는 맞지 않습니다. 퓨어스토리지의 테스트 결과는 다음과 같습니다.

데이터 세트가 지정된 스토리지 크기에 맞게 트랜스코딩될 때 성능이 저하됩니다.

그러나 절충점을 찾으려면 다양한 트랜스코딩 설정을 평가하는 데 계산 비용이 많이 듭니다. 실험에서는 JPEG 트랜스코딩에 30가지의 압축 강도 값을 사용했고, H.264/AVC 트랜스코딩에는 30가지 이상의 GOP 및 CRF 설정을 사용했습니다. 또한 트랜스코딩의 파라미터와 인코더는 다양하기 때문에 모든 것을 다루지는 못했습니다. 따라서 결과는 현재 실험 설정으로 제한되었습니다.

VII. 결론

본 논문에서는 Kinet- ics400을 이용한 실험을 통해 영상 품질과 동작 인식 모델의 성능 간의 관계를 정량적으로 분석했습니다. JPEG 트랜스코딩을 사용한 실험에서는 압축 강도 cs가 육안으로 화질 저하가 관찰되지 않는 범위인 70 이하에서는 심각한 성능 저하가 나타나지 않았으며, cs가 80 이상에서는 화질 지수에 비례하여 선형적으로 성능이 저하되는 것을 확인할 수 있었습니다. H.264/AVC 트랜스코딩 실험 결과, 파일 크기를 약 30% 줄일 수 있는 CRF 30에서도 큰 성능 저하(최대 -1%)가 없는 것으로 나타났습니다. 사전 학습된 모델과 트랜스코딩된 비디오로 학습된 모델을 사용한 평가에서도 동일한 결과가 관찰되었습니다.

향후 작업에는 SSv2 [11], HVU [9], Moments in Time [10] 등 트리밍된 동영상의 다른 대규모 데이터 세트와 트리밍되지 않은 동영상 데이터 세트 [8]를 사용하여 유사한 경향을 관찰할 수 있는지 조사하고, H.265/HEVC와 같은 최신 인코더를 사용하는 것도 포함됩니다.

참고 자료

[1] M. Vrigkas, C. Nikou 및 I. Kakadiaris, "인간 활동 인식 방법의 검토," *로봇 공학 및 인공 지능의 프론티어*, 2권, 11 년 2015 월.

[2] M. S. 허친슨과 V. N. 가데팔리, "비디오 액션 이해," *IEEE 액세스*, 9권, 134 611-134 637쪽, 2021.

[3] Y. Zhu, X. Li, C. Liu, M. Zolfaghari, Y. Xiong, C. Wu, Z. Zhang, J. Tighe, R. Manmatha 및 M. Li, "심층 비디오 동작 인식에 대한 포괄적인 연구", *CoRR*, vol. [온라인]. 이용 가능: https://arxiv.org/abs/2012.06567

[4] J. 셀바, A. S. 요한슨, S. 에스칼레라, K. 나스롤라히, T. B. 모슬룬드, 및 A. Clapés, "비디오 트랜스포머: 설문 조사", *CoRR*, vol. abs/2201.05991, 2022. [온라인]. 이용 가능: https://arxiv.org/abs/2201.05991

[5] W. Kay, J . Carreira, K . Simonyan, B . Zhang, C . Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman 및 A. Zisserman, "역학 인간 행동 비디오 데이터 세트," *CoRR*, vol. [온라인]. 이용 가능: http://arxiv.org/abs/1705.06950

[6] C. Gu, C. Sun, S. Vijayanarasimhan, C. Pantofaru, D. A. Ross, G. Toderici, Y. Li, S. Ricco, R. Sukthankar, C. Schmid, 및 J. Malik, "AVA: 시공간적으로 국지화된 원자 시각적 동작의 비디오 데이터 세트," *CoRR*, vol. [온라인]. 이용 가능: http://arxiv.org/abs/1705.08421

[7] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, "Youtube-8m: 대규모 비디오 분류 벤치마크", *CoRR*, vol. [온라인]. Available: http://arxiv.org/abs/1609.08675

[8] B. G. Fabian Caba Heilbron, Victor Escorcia, J. C. Niebles, "Activi- tynet: 인간 활동 이해를 위한 대규모 비디오 벤치마크", *IEEE 컴퓨터 비전 및 패턴 인식 컨퍼런스 논문집*, 2015, 961-970쪽.

[9] A. 디바, M. 파야즈, V. 샤르마, M. 팔루리, J. 갈, R. 스티펠하겐, 및 L. Van Gool, "대규모 전체론적 비디오 이해", *유럽 컴퓨터 비전 컨퍼런스*. Springer, 2020, 593-610쪽.

[10] M. 몬포트, A. 안도니안, B. 저우, K. 라마크리슈난, S. A. 바갈, T. Yan, L. Brown, Q. Fan, D. Gutfruend, C. Vondrick *외*, "순간 데이터 세트: 이벤트 이해를 위한 100만 개의 동영상", *IEEE Trans- actions on Pattern Analysis and Machine Intelligence*, 1-8쪽, 2019.

[11] R. Goyal, S. Ebrahimi Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax 및 R. Memisevic, "시각적 상식을 학습하고 평가하기 위한 '무언가' 비디오 데이터베이스," *IEEE 국제 컴퓨터 비전 컨퍼런스 (ICCV) P r o c e e d i n g s* , Oct 2017.

[12] K. Soomro, A. R. Zamir 및 M. Shah, "UCF101: 야생의 비디오에서 얻은 101개의 인간 행동 클래스 데이터 세트", *CoRR*, vol. [온라인]. 이용 가능: http://arxiv.org/abs/1212.0402

[13] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "Hmdb: 인간 동작 인식을 위한 대규모 비디오 데이터베이스," *2011 국제 컴퓨터 비전 컨퍼런스*, 2011, 2556-2563쪽.

[14] C. Feichtenhofer, "X3d: 효율적인 비디오 인 식 을 위한 아키텍처 확장," *IEEE/CVF 컴퓨터 비전 및 패턴 인식 컨퍼런스(CVPR) 논문집*, 2020년 6월.

[15] C. Feichtenhofer, H. Fan, J. Malik, K. He, "비디오 인식을 위한 슬로우패스트 네트워크", *IEEE/CVF 국제 컴퓨터 비전 학술대회(ICCV) 논문집*, 2019년 10월.

[16] K. Hara, H. Kataoka, Y. Satoh, "시공간적 3D CNN이 2D CNN과 이미지넷의 역사를 되 짚 어 볼 수 있을까요?" *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, 6546-6555쪽.

[17] G. Bertasius, H. Wang, L. Torresani, "시공간 주의만 있으면 비디오 이해에 필요한 전부인가?", *38회 국제 기계 학습 컨퍼런스 논문집(Proceedings of the 38th International Conference on Machine Learning*, ser. 기계 학습 연구 절차, M. Meila 및 T. Zhang, Eds. 139. PMLR, 2021 년 7 월 18-24 일, 813-824 쪽. [온라인]. 이용 가능: https://proceedings.mlr.press/v139/bertasius21a.html

[18] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin 및 H. Hu, "비디오 스윈 변압기," *CoRR*, vol. [온라인]. Available: https://arxiv.org/abs/2106.13230

[19] S. Srivastava, G. Ben-Yosef, 및 X. Boix, "심층 신경망의 최소 이미지: 자연 이미지에서 깨지기 쉬운 물체 인식", *국제 학습 표현 컨퍼런스*, 2019. [온라인]. Available: https://openreview.net/forum?id=S1xNb2A9YX

[20] J. Seo and H. Park, "Deep Collaborative Learning을 이용한 초저해상도 이미지에서의 객체 인식",*IEEE Access*, vol.

[21] Z. Wang, S. Chang, Y. Yang, D. Liu 및 T. S. Huang, "딥 네트워크를 이용한 초저해상도 인식 연구", *2016 IEEE 컴퓨터 비전 및 패턴 인식 컨퍼런스 (CVPR)*, 2016, 4792-4800쪽.

[22] D. Cai, K. Chen, Y. Qian, and J.-K. Kämäräinen, "컨볼루션 저해상도 세분화 분류", *Pattern Recognition Letters*, vol. 119, 03 2017.

[23] D. Hendrycks 및 T. Dietterich, "일반적인 손상 및 섭동에 대한 신경망 견고성 벤치마킹", *학습 표현에 관한 국제 컨퍼런스*, 2019. [온라인]. Available: https://openreview.net/forum?id=HJz6tiCqYm

[24] S. F. Dodge와 L. J. Karam, "이미지 품질이 심층 신경망에 미치는 영향에 대한 이해", *CoRR*, vol. [온라인]. 이용 가능: http://arxiv.org/abs/1604.04004

[25] M. Hou, S. Liu, J. Zhou, Y. Zhang, Z. Feng, "초고해상도 지향 생성 적대 네트워크를 사용한 극저해상도 활동 인식", *Micromachines*, 12권 6호, 2021. [온라인]. Available: https://www.mdpi.com/2072-666X/12/6/670

[26] U. Demir, Y. S. Rawat, and M. Shah, "Tinyvirat: 저해상도 비디오 동작 인식", *2020 제25회 국제 패턴 인식 컨퍼런스(ICPR)*, 2021, 7387-7394쪽.

[27] A. B. Jung, K. Wada, J. Crall, S. Tanaka, J. Graving, C. Reinders, S. 야다브, J. 바네르지, G. 벡세이, A. 크래프트, Z. 루이, J. 보로벡, C. 발렌틴, S. Zhydenko, K. Pfeiffer, B. Cook, I. Fernández, F.-M. 드 레인빌, C.-H. Weng, A. Ayala-Acevedo, R. Meudec, M. Laporte *외*, "imgaug", https://github.com/aleju/imgaug, 2020, 온라인; 01-Feb-2020에 액세스했습니다.

[28] M. Singh, S. Nagpal, R. Singh, M. Vatsa, "초저해상도 이미지 인식을 위한 이중 지향 캡슐 네트워크," *CoRR*, vol. [온라인]. Available: http://arxiv.org/abs/1908. 10027

[29] ... "(대우) 저해상도 이미지 분류를 위한 파생넷", *IEEE 트랜잭션에 관한 패턴 분석 및 기계 지능*, 1-1쪽, 2021.

[30] S. Shekhar, V. M. Patel, 및 R. Chellappa, "합성 기반의 강력한 저해상도 얼굴 인식", *CoRR*, vol. [온라인].
이용 가능: http://arxiv.org/abs/1707.02733

[31] A. Azulay와 Y. Weiss, "심층 컨볼루션 네트워크는 왜 작은 이미지 변환에 그렇게 잘 일반화되지 않는가?" *기계 학습 연구 저널*, 20, 184, 1-25 쪽, 2019. [온라인]. Available: http://jmlr.org/papers/v20/19-519.html

[32] S. 카라한, M. K. 일디림, K. 키르타즈, F. S. 렌데, G. 부툰, 및 H. K. 에케넬, "이미지 품질 저하는 심층 신경망 기반 얼굴 인식에 어떤 영향을 미칩니까?" *CoRR*, vol. abs/1608.05246, 2016. [온라인]. 이용 가능: http://arxiv.org/abs/1608.05246

[33] K. Grm, V. Štruc, A. Artiges, M. Caron, H. Ekenel, "이미지 품질 저하에 대한 얼굴 인식을 위한 딥러닝 모델의 강점과 약점 ," *IET Biometrics*, 7권 , 09 2017.

[34] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, "Imagenet: 대규모 계층적 이미지 데이터베이스", *2009 IEEE Conference on 컴퓨터 비전 및 패턴 인식*, 2009, 248-255페이지.

[35] Y. Wang, Y. Cao, Z.-J. Zha, J. Zhang, Z. Xiong, "저화질 이미지 분류를 위한 심층 성능 저하 사전 처리", *IEEE/CVF 컨퍼런스(컴퓨터 비전 및 패턴 인식(CVPR)*, 2020년 6월).

[36] D. Liu, B. Cheng, Z. Wang, H. Zhang, T. S. Huang, "딥 네트워크를 통한 불리한 조건에서의 시각 인식 향상", *IEEE Transactions on Image Processing*, 28, 9권, 4401-4412쪽, 2019.

[37] J. 참조 및 S. Rahman, "인간 행동 인식에서 낮은 비디오 품질이 미치는 영향", *2015 국제 디지털 이미지 컨퍼런스 컴퓨팅: 기술 및 응용(DICTA)*, 2015, 1-8쪽.

[38] R. Pourreza, A. Ghodrati, 및 A. Habibian, "압축 비디오 인식: 도전과제와 약속," *IEEE/CVF 국제 컴퓨터 비전 컨퍼런스(ICCV) 워크샵 (Proceedings of the IEEE/CVF In-*), 2019년 10월.

[39] S. Tomar, "ffmpeg로 비디오 포맷 변환하기," *Linux Journal*, vol. 2006, 146호, 10면, 2006.

[40] S. H. 칸, M. 나세르, M. 하야트, S. W. 자미르, F. S. 칸, 및 M. 샤, "시각의 변압기: 설문 조사", *CoRR*, vol. [온라인]. 이용 가능: https://arxiv.org/abs/2101.01169

[41] H. Fan, T. Murrell, H. Wang, K. Alwala, Y. Li, Y. Li, B. Xiong, N. Ravi, M. Li, H. Yang, J. Malik, R. Girshick, M. Feiszli, A. Adcock, W.-Y. Lo, 및 C. Feichtenhofer, "Pytorchvideo: 비디오 이해를 위한 딥 러닝 라이브러리 ", 10 2021, 3783-3786쪽.

[42] D. P. Kingma와 J. Ba, "Adam: 확률적 최적화를 위한 방법," *제3회 학습 표현에 관한 국제 컨퍼런스, ICLR 2015, 미국 캘리포니아주 샌디에이고, 2015년 5월 7-9일, 컨퍼런스 트랙 프로시딩*, Y. Bengio와 Y. LeCun, Eds., 2015. [온라인]. Available: http://arxiv.org/abs/1412.6980

오타니 아오이는 2022년 나고야 공과대학에서 학사 학위를 받았습니다. 그의 연구 관심 분야는 컴퓨터 비전과 동작 인식입니다.

료타 하시구치는 2022년 나고야 공과대학에서 학사 학위를 받았습니다. 컴퓨터 비전과 행동 인식에 관심이 있습니다.

카즈키 오미는 2022년 나고야 공과대학교에서 학사 학위를 받았습니다. 그의 연구 분야는 컴퓨터 비전과 동작 인식입니다.

노리시게 후쿠시마는 2004년, 2006년, 2009년에 일본 나고야대학교에서 각각 학사, 석사, 박사 학위를 받았습니다. 2009년에는 조교수로, 2015년에는 일본 나고야 공과대학의 부교수로 부임했습니다. 그의 연구 관심 분야는 이미지 신호 처리, 병렬 이미지 처리 및 컴파일러입니다. 그는 IEICE, IPSJ, IEEE(CAS, SPS)의 회원입니다.

토루 타마키는 1996년, 1998년, 2001년에 각각 일본 나고야대학교에서 정보공학 학사, 석사, 박사 학위를 받았습니다. 일본 니가타 대학 조교수, 일본 히로시마 대학 부교수를 거쳐 현재 일본 나고야 공과대학 컴퓨터과학과 교수로 재직 중입니다.

2015년에는 프랑스 파리 ESIEE에서 부연구원을 역임했습니다. 그의 연구 관심 분야는 다음과 같습니다.

컴퓨터 비전, 이미지 인식, 머신 러닝, 의료 이미지 분석 등 다양한 분야에서 활약하고 있습니다.