

CAN-bus 및 스마트폰 센서 데이터 융합을 통한 운전자 식별 연구

이가현¹, 정유빈², 이기훈¹, 허종욱^{1*}

¹한림대학교 컴퓨터공학과 석사과정

²한림대학교 빅데이터학과 학사

^{1*}한림대학교 컴퓨터공학과 교수

gh.lee.hallym@gmail.com, jungyoubin52@gmail.com, drlgns1104@naver.com, juhou@hallym.ac.kr

A Study on Driver Identification Using CAN-bus and Smartphone Sensor Data Fusion

Ka-Hyun Lee¹, Yu-been Jeong², Gi-Hun Lee¹, Jong-Uk Hou^{1*}

^{1,2}Dept. of Computer Engineering, Hallym University

^{1*}Dept. of Computer Engineering, Hallym University

요 약

차량의 증가와 함께 교통 안전에 대한 관심이 높아지면서, 운전자 프로파일링에 대한 필요성이 증대되고 있다. 또한, 자율주행 기술의 발전과 자동차 보안 문제의 증가로 운전자 식별 연구의 중요성도 커지고 있다. 이에 본 연구는 차량의 CAN-bus 데이터와 스마트폰 센서 데이터를 직접 수집하여 운전자를 식별하는 새로운 방법을 제안한다. 기존의 생체 인식 정보나 운전자의 이미지를 활용하는 방법과 달리, 본 연구는 주행 중 수집된 정량적 데이터를 이용해 프라이버시 위험을 최소화하였다. 특히, 다양한 주행 상황에서 직접 수집한 CAN-bus 및 센서 데이터를 기반으로 한 멀티모달 퓨전 기법을 통해 운전자의 행동 및 반응을 정밀하게 분석하였다. 실험 결과, 제안된 방법이 기존 방법보다 정확하게 운전 습관을 식별하였으며, 운전자 식별의 정확성을 높여 미래의 운전자 프로파일링 연구에 기여할 수 있을 것으로 기대된다.

1. 서론

차량의 수가 급격히 증가함에 따라 교통 안전과 차량 보안 문제가 심화되고 있다. 교통사고, 차량 도난 [1]과 같은 위협은 운전자와 차량의 안전을 보장하기 위해 보다 정교한 솔루션을 필요로 한다. 이러한 문제를 해결하기 위해 운전자 식별 기술의 중요성이 부각되고 있다[2]. 운전자 식별이란 개별 운전자의 주행 패턴과 특성을 분석하여 특정 운전자를 정확하게 구별하는 기술로, 이를 통해 차량 보안 강화, 맞춤형 서비스 제공, 사고 예방 등을 기대할 수 있다.

기존 연구들은 주로 CAN-bus 데이터[3]나 스마트폰 센서 데이터[4]를 활용하여 운전자의 행동을 분석해왔다. CAN-bus 데이터는 차량의 물리적 상태와 운전자의 조작 행동을 반영하고, 스마트폰 센서 데이터는 운전자의 움직임과 주행 환경에 대한 정보를 제공한다. 하지만 개별 데이터 소스만으로는 실시간으로 변

화하는 도로 상황에서의 즉각적인 운전자 반응을 충분히 포착하기에 한계가 있다. 예를 들어, CAN-bus 데이터만으로는 차량의 상태 변화를 확인할 수 있지만, 운전자가 환경 변화에 어떻게 반응하는지에 대한 세부적인 정보는 부족할 수 있다. 반대로, 스마트폰 센서 데이터는 외부 움직임을 포착할 수 있지만 차량 내부의 상태 변화까지 반영하는 데에는 한계가 있다.

본 연구에서는 이러한 한계를 극복하기 위해 CAN-bus 데이터와 스마트폰 센서 데이터를 융합하여 운전자를 식별하는 방법을 제안한다. 이 두 가지 데이터 소스는 상호보완적인 역할을 하여, 개별 데이터의 약점을 보완하고 다양한 상황에서 운전자의 반응을 정확히 분석할 수 있는 가능성을 높인다. 특히, 다양한 주행 상황에서 직접 수집한 데이터로 운전자의 주행 패턴과 반응을 실시간으로 더 정확하게 분석할 수 있도록 설계되었다. 실험을 통해 제안된 방법이 기존 방법보다 높은 정확도로 운전자를 식별하며, 교통 안전과 차량 보안에 기여할 수 있음을 입증하였다.

* 교신저자 (corresponding author)

2. 데이터셋 수집

본 연구에서는 Tesla Model 3 차량을 이용하여 CAN-bus 데이터와 스마트폰 센서 데이터를 수집하였다. 15 명의 운전자가 세 가지 주행 코스(그림 1)를 각 4 회씩 주행하며 데이터를 수집하였으며, 각 코스의 주행 시간은 약 15 분이다. 주행 코스는 8~9 월, 1~3 월내의 도심에서 다양한 도로 조건과 환경을 포함하도록 설계되어, 여러 주행 상황에서의 데이터를 확보할 수 있었다. 데이터 수집 환경은 그림 2 에 제시된 Tesla 차량과 스마트폰 및 CAN-bus 장비 설치 환경으로 구성되었다.

Kvaser CAN-bus 장비를 사용하여 차량의 내부 통신 네트워크에서 실시간으로 차량 속도, 가속도, 핸들 각도, 브레이크 상태 등의 데이터를 추출하였다. 수집된 CAN-bus 데이터의 총 용량은 279.7 GB 이다. 동시에, 스마트폰 내장 센서(가속도계, 자이로스코프, GPS 등)를 활용하여 운전 중 발생하는 움직임과 위치 데이터를 기록하였다. 스마트폰 센서 데이터는 총 7.2GB 가 수집되었다. 모든 운전자는 동일한 조건 하에 데이터를 제공하였으며, 데이터 수집 장비 또한 동일한 방식으로 설치 및 작동되었다. 데이터는 익명화 처리되어 개인정보 보호가 이루어졌다.



(그림 1) 주행 코스 지도



(그림 2) 데이터 수집 환경
(좌) 차량 (우) 스마트폰 센서 및 CAN-bus 장비

3. 방법론

3.1. CAN-bus 데이터 기반 Classifier

본 섹션에서는 CAN-bus 데이터를 기반으로 운전자를 식별한다. Kwak [5]의 방법론을 참조하여, 표 3 의 C1 부터 C7 까지 7 개의 주요 신호를 선택해 60 초 간격의 윈도우와 Wavelet Transform 을 적용하여 주행 특성을 분석하였다. 이 과정에서 엔트로피, 백분위수, 평균, 표준편차 등의 통계적 특징을 추출하였다.

추출된 특징들은 SVM(Support Vector Machine) 알고리즘에 입력되어 운전자를 식별하기 위한 학습에 활용되었다. SVM 은 각 운전자의 고유한 주행 패턴을 학습하고, 이를 바탕으로 높은 정확도로 운전자를 구별할 수 있도록 설계되었다. 실험에 사용된 CAN-bus 데이터의 규격은 표 4 에 표시되어 있다.

3.2. 스마트폰 센서 데이터 기반 Classifier

본 섹션에서는 표 3 의 S1 과 S4 데이터를 이용하여 운전자를 식별한다. Sánchez [7]의 방법론을 참조하여, 주행 중 기록된 센서 데이터를 35 초 간격으로 분할한 후, 각 분할된 데이터를 STFT(Short-Time Fourier Transform) 기반으로 스펙트로그램을 생성한다. 생성된 스펙트로그램은 로그 스케일로 변환한 후, min-max normalization 을 거쳐 입력 데이터로 사용된다. 실험에 사용된 스마트폰 센서 데이터의 규격은 표 4 에 표시되어 있다.

스마트폰 데이터 인코더는 ResNet50 과 GRU 네트워크를 결합하여 운전자의 행동 패턴을 학습한다. ResNet50 은 스펙트로그램으로부터 공간적 특징을 추출하고, GRU 는 시간적 패턴을 학습하여 운전자의 고유한 주행 스타일을 파악한다. 이 인코더는 운전 중의 미세한 움직임과 변화 패턴을 학습함으로써, 높은 정확도로 운전자를 구별할 수 있도록 설계되었다.

3.3. 멀티모달 데이터 융합

본 섹션에서는 CAN-bus 데이터와 스마트폰 센서 데이터를 결합하여 운전자 식별의 성능을 향상시키기 위해 데이터 융합 기법을 적용하였다. (그림 3 참조) 두 데이터 소스의 분류기는 각각의 특성에 따라 개별적으로 학습된 후, 앙상블 기법을 통해 결합되었다. 이때 가중치 앙상블 방법을 사용하여 두 분류기의 출력 확률을 최종적으로 결합하였다.

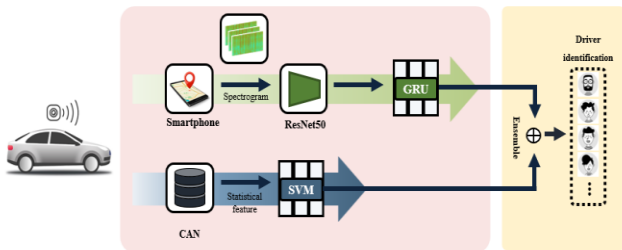
각 분류기의 출력 확률은 클래스에 대해 각 데이터 소스에서 운전자를 정확하게 식별할 확률을 나타낸다. 앙상블 예측 확률은 개별 확률을 가중합한 값으로 계산된다. 이때 각 데이터 소스가 최종 예측에 기여하는 비율을 나타내는 가중치는 실험을 통해 최적화되어 균등하게 설정된다.

Method	Feature(duration)	voting-1				voting-3				voting-10				voting-all			
		ACC	Top3	mAP	F1	ACC	Top3	mAP	F1	ACC	Top3	mAP	F1	ACC	Top3	mAP	F1
회차별 교차검증																	
[5]	C (60sec)	32.70	67.85	25.10	32.23	44.53	77.28	38.30	43.20	60.38	87.43	57.35	58.03	78.35	94.98	82.53	72.50
[6]	C (60sec)	21.80	49.60	13.33	20.80	27.45	58.50	19.48	26.15	35.23	67.18	28.95	33.28	53.33	85.00	63.90	45.13
[7]	S (35sec)	36.45	65.05	97.48	36.45	44.75	76.63	95.15	44.75	54.55	83.68	90.6	54.55	63.35	90.00	83.70	63.35
[8]	S (90sec)	44.8	70.6	51.3	42.1	48.5	73.2	56.8	45.2	49.2	78.6	68.6	43.7	60.4	77.2	76.8	52.1
CS fusion (Ours)	C, S	-	-	-	-	-	-	-	-	-	-	-	-	86.68	93.33	82.48	89.45
코스별 교차검증																	
[5]	C (60sec)	28.07	60.97	20.53	27.47	33.93	68.37	32.17	33.20	41.30	76.80	48.53	40.40	51.10	84.47	75.47	43.13
[6]	C (60sec)	19.60	46.17	10.83	18.40	23.30	53.60	15.30	21.73	27.63	60.13	22.77	25.13	40.00	75.57	55.50	29.23
[7]	S (35sec)	35.8	68.5	95.87	35.8	42.53	76.03	95.63	42.53	49.87	82.7	94.97	49.87	64.43	95.57	98.53	64.43
[8]	S (90sec)	22	47.1	27.8	17	22.4	49.8	32.8	16.5	22.3	51	45	14.3	23.9	50.7	61.3	11.5
CS fusion (Ours)	C, S	-	-	-	-	-	-	-	-	-	-	-	-	75.57	97.77	69.17	86.27

<표 1> 회차별, 코스별 교차검증에 대한 비교실험과 메인실험 결과

Method (feature)	ACC	Top3	mAP	F1
회차별 교차검증				
CAN-based (C)	78.4	95	82.5	72.5
Sensor-based (S)	63.4	90	83.7	63.4
CS-fusion (C, S)	86.7	93.3	82.5	89.5
CS-Weighted-fusion (C, S)	90	95	87.8	90.6
코스별 교차검증				
CAN-based (C)	51.1	84.5	75.5	43.1
Sensor-based (S)	64.4	95.6	98.5	64.4
CS-fusion (C, S)	75.6	97.8	69.2	86.3
CS-Weighted-fusion (C, S)	77.5	97.1	71.6	86.2

<표 2> ALL Voting 기준 데이터 융합에 대한 결과



(그림 3) 제안하는 모델 아키텍처

4. 비교실험

CAN-bus 와 스마트폰 센서 데이터를 활용한 운전자 식별 성능을 평가하기 위해 [5]부터 [8]까지의 기존 모델들과 비교 실험을 진행하였다. [5]와 [7]은 본 연구에서 제안한 앙상블 기법에 사용된 방법론이며, [6]과 [8]은 비교를 위한 기존 연구 모델이다.

표 1 과 표 2 의 모든 실험은 교차검증으로 진행되었으며, 각 회차 또는 코스를 validation 세트와 사용 하고, 나머지 데이터를 train 세트로 활용하여 학습하였다. (표 4 의 데이터 세트별 비율 참조)

4.1. CAN-bus 데이터 기반 비교실험

Kwak [6]의 방법론을 참조하여, 표 3 의 C1 부터 C7 까지 주요 CAN-bus 신호(그림 3 참조)를 선택하였다. 각 신호는 0 에서 1 로 정규화되었으며, 운전 패턴의 통계적 특성을 추출하기 위해 60 초 간격으로 평균, 중앙값, 표준 편차 등의 통계적 특징을 계산하였다. 이렇게 처리된 피쳐 벡터를 기반으로 KNN 알고리즘을 사용해 운전자 식별 모델을 학습하였다.

ID ¹	Feature name	Description
C1	DL_vehicleSpeed	차량 속도
C2	SteeringAngle129	스티어링 휠 각도
C3	SteeringSpeed129	스티어링 휠 회전 속도
C4	DL_regenLight	브레이크 재생 표시등
C5	DIR_torqueActual	실제 토크 값
C6	DIR_torqueCommand	토크 명령 값
C7	RearTorque1D8	실제 후륜 토크
C8	RearHeatPower266	후방 열 출력
C9	DL_accelPedalPos	운전자의 가속 페달 압력
C10	SmoothBattCurrent132	고전압 배터리 전류
C11	VCFRONT_chillerExvFlowm3	VCFRONT 냉각 시스템의 유량
C12	BMS_maxDischargePower	BMS의 최대 방전 전력
C13	SystemHeatPowerMax268	최대 시스템 열 출력
C14	VCFRONT_pumpBatteryRPM	VCFRONT 펌프 배터리의 실제 회전수 (RPM)
C15	BattVoltage132	배터리 전압
C16	SystemHeatPower268	시스템 열 출력
C17	RearMotorCurrent126	후륜 모터 전류
C18	SmoothBattCurrent	고전압 배터리의 부드러운 전류
C19	RearTorqueRequest1D8	차량 후륜 토크 요청
C20	RearPower266	후방 출력
C21	DL_uiSpeed	UI 속도
C22	MinVoltage2D2	최소 배터리 전압
C23	MaxVoltage2D2	최대 배터리 전압
C24	DIR_axleSpeed	차축 속도
S1	Accelerometer	가속도 측정
S2	Speed	속도 데이터
S3	Orientation	방향 데이터
S4	Gyroscope	자이로스코프 측정
S5	GPS	위치 데이터

<표 3> 수집한 데이터셋 피쳐의 디테일
(¹ID 의 'C' 와 'S' 는 각각 CAN-bus 데이터와 스마트폰 센서 데이터를 의미함.)

Course Partition	# Examples	Round Partition	# Examples
Course A	1,092 (35%)	Round 1	773 (25%)
Course B	822 (27%)	Round 2	770 (25%)
Course C	1,175 (38%)	Round 3	773 (25%)
		Round 4	773 (25%)

Total: 3,089

Course Partition	# Examples	Round Partition	# Examples
Course A	35,645 (36%)	Round 1	25,156 (25%)
Course B	25,791 (26%)	Round 2	25,052 (25%)
Course C	38,535 (38%)	Round 3	25,124 (25%)
		Round 4	24,819 (25%)

Total: 100,151

<표 4> 학습에 사용한 데이터셋 개수
(상) 스마트폰 센서 데이터 (하) CAN-bus 데이터

4.2. 스마트폰 센서 데이터 기반 비교실험

Rouhollah [8]의 연구 방법을 참조하여, 스마트폰의 S1, S4 데이터를 선택하였다 (표 3 참조). 가속도계와 자이로스코프계의 x, y, z 축 데이터를 기반으로 하며, 90 초의 윈도우 크기와 45 초의 오버랩을 사용하여 데이터를 세분화하였다. 각 세그먼트에 대해 히스토

그램 피처를 추출하고, 이를 바탕으로 운전자 식별을 위한 특징 벡터를 구성하였다. 또한, 데이터 증강 기법으로서 GAN 을 사용하여 데이터의 다양성을 높였다. 모델의 학습과 테스트 과정에서는 SVC, MLP, KNN, 그리고 Random Forest 분류기들을 사용한 스택킹 분류기를 통해 모델의 성능을 최적화하였다.

CAN-bus 데이터와 스마트폰 센서 데이터를 모두 평가하기 위해 사용한 주요 평가지표는 다음과 같다:

- Voting: 하나의 샘플로 예측(Voting 1), 3 개의 연속된 샘플로 예측(Voting 3), 10 개의 샘플로 예측(Voting 10), 모든 샘플을 활용한 예측(Voting all) 방식으로 평가한다. (표 1 의 duration 참조.)
- Top-3: 모델이 상위 3 개의 예측 중 실제 운전자를 포함하는 빈도를 평가하는 지표이다.
- mAP(Mean Average Precision): 정밀도와 재현율을 다양한 임계값에서 평균하여 평가한다.
- F1: 정밀도와 재현율의 조화 평균으로, 두 지표의 균형을 유지하며 모델의 전반적인 성능을 평가한다.

4.3. 실험 결과 및 고찰

본 연구에서 제안한 CAN-bus 와 스마트폰 센서 데이터 융합(CS Fusion) 기법은 표 1 과 표 2 에서 보이듯이, 단일 데이터 기반 방법에 비해 전반적으로 더 높은 성능을 보여주었다. 특히, Voting-all 기준에서 CS Fusion 방법은 기존 방법들에 비해 ACC, F1 지표에서 크게 향상된 결과를 나타냈다. 이러한 결과는 CAN-bus 데이터가 차량 제어 신호와 관련된 정보를 제공하고, 스마트폰 센서 데이터가 미세한 움직임과 외부 환경 변화를 포착함으로써 상호보완적으로 작용했기 때문으로 분석된다.

또한, 표 2 에서 CS Fusion 방법은 모든 교차검증에서 기존 방법에 비해 성능이 우수하다. 특히, 회차별 검증에서 Weighted-Fusion 기법이 ACC 90%로 가장 높은 성능을 기록하였다. 코스별 검증에서도 ACC 77.5%, F1 86.2%로 뛰어난 성능을 보였는데, 이는 다양한 주행 환경에서도 일관되게 높은 정확성을 유지할 수 있음을 보여준다. 이러한 결과는 데이터 융합 기법이 단일 데이터 소스에서 발생할 수 있는 정보 손실을 보완하고, 실시간 운전자 식별의 정확성을 높이는 데 기여할 수 있음을 시사한다.

5. 결론

본 연구는 CAN-bus 데이터와 스마트폰 센서 데이터를 결합한 멀티모달 데이터 융합 기법을 활용하여 운전자를 식별하는 방법을 제안하였다. 각각의 데이터 소스를 독립적으로 학습한 후, 앙상블 기법을 통해 예측 성능을 향상시켰다. CAN-bus 데이터는 차량 제어 신호를, 스마트폰 센서 데이터는 미세한 움직임을 포착하여 운전자를 높은 정확도로 식별할 수 있었다. 멀티모달 데이터 융합 기법은 단일 데이터 소스의 한계를 보완해 운전자의 주행 습관을 더 정밀하게 분석하고, 차량 보안과 맞춤형 서비스에서 우수

한 정확성을 제공한다.

향후 연구에서는 더 다양한 데이터 소스를 융합하여 운전자 식별의 정확도를 더욱 향상시키는 방안을 모색할 것이다. 본 연구에서 수집된 데이터셋은 데이터 융합의 기초 자료로 활용될 수 있으며, 이를 통해 운전자 식별 기술을 한층 발전시킬 수 있다. 특히, 자율주행 시스템에서 운전 주체를 명확히 구분함으로써, 사고 발생 시 책임 소재를 명확히 규명하는 데 중요한 역할을 할 것으로 기대된다. 이러한 연구는 스마트 모빌리티 기술의 보안을 강화하고, 자율주행 시스템의 안정성을 증대시키는 데 기여할 것이다.

사사

이 논문은 2022 년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.2022R1A4A1033600).

참고문헌

- [1] National Insurance Crime Bureau, "Vehicle Thefts Surge Nationwide in 2023," [Online]. Available: <https://www.nicb.org/news/news-releases/vehicle-thefts-surge-nationwide-2023>. Accessed: April 12, 2024.
- [2] Rahim, Mussadiq Abdul, et al., "Zero-to-stable driver identification: A non-intrusive and scalable driver identification scheme," *IEEE Transactions on Vehicular Technology*, 69(1), pp. 163-171, 2019.
- [3] Fugiglando, Umberto, et al., "Driving behavior analysis through CAN bus data in an uncontrolled environment," *IEEE Transactions on Intelligent Transportation Systems*, 20(2), pp. 737-748, 2018.
- [4] Chan, Teck Kai, et al., "A comprehensive review of driver behavior analysis utilizing smartphones," *IEEE Transactions on Intelligent Transportation Systems*, 21(10), pp. 4444-4475, 2019.
- [5] Kwak, Byung Il, Mee Lan Han, and Huy Kang Kim, "Driver identification based on wavelet transform using driving patterns," *IEEE Transactions on Industrial Informatics*, 17(4), pp. 2400-2410, 2020.
- [6] Kwak, Byung Il, JiYoung Woo, and Huy Kang Kim, "Know your master: Driver profiling-based anti-theft method," *2016 14th Annual Conference on Privacy, Security and Trust (PST)*, Auckland, New Zealand, 2016, pp. 129-136.
- [7] Sánchez, Sara Hernández, Rubén Fernández Pozo, and Luis Alfonso Hernández Gómez, "Driver identification and verification from smartphone accelerometers using deep neural networks," *IEEE Transactions on Intelligent Transportation Systems*, 23(1), pp. 97-109, 2020.
- [8] Ahmadian, Rouhollah, Mehdi Ghatee, and Johan Wahlström, "Discrete wavelet transform for generative adversarial network to identify drivers using gyroscope and accelerometer sensors," *IEEE Sensors Journal*, 22(7), pp. 6879-6886, 2022.