1. 서론

EEG(Electroencephalography)는 뇌에서 발생하는 신경생리학적 반응을 기록하는 기법으로 20세기 후반에 등장했다[1]. BCI(Brain-Computer-Interface)는 신체적 움직임 없이 뇌파만으로 기계나 컴퓨터를 제어하는 기술이다. EEG 최근 뇌파를 간편하게 측정할 수 있는 장치들이 개발되면서, BCI를 의학, 문화, 교육, 국방 등 과 같은 다양한 분야에서 활용하기 위한 연구들이 진행되고 있다. 의료 분야에서는 부상 예방에서부터 심각한 부상에 대한 신경 재활에 대해 활용 가능하다[2]. 생각을 읽고, 원격 소통할 수 있는 기술은 게임과 엔터테인먼트뿐만 아니라 교육, 생산, 마케팅, 보안 등 수많은 분야에서 사용될 수 있다[2]. BCI 기술 개발을 위해서는 BCI에 사용되는 EEG 데이터의 정확한 분석이 필수적이다. 뇌는 신체의 모든 부분에 관여하는 신체 기관이기 때문에 신체에서 발생하는 다양한 신호에 영향을 받는다. 예를 들어 눈동자 움직임에 의한 안전도 신호(EOG : Electrooculography), 근육의 수축, 이완에 의한 근전도 신호(EMG : Electromyography), 심장 박동에 의한 심전도(ECG : Electrocardiography) 신호가 이에 포함된다. EEG 데이터를 분석하기 위해서는 각각의 세부적인 처리 목적에 따라 특정 주파수의 EEG를 분리하고, 특징을 추출해 내는 등 다양한 전처리 과정이 필요하다. 또한 전처리를 거친 EEG를 보다 정확하게 분석하기 위해서는 EEG의 특성을 잘 반영할 수 있는 분석 모델과 연동이 필요하다. 이 논문에서는 EEG 신호의 종류 중 동작을 상상하는 상황에서 나타나는 EEG 신호를 분석하는 연구에 초점을 맞추었다.

최근 Motor Imagery EEG를 분류하기 위해 CNN이나 LSTM, Resnet 등의 여러 알고리즘들을 사용하는 방법들이 연구되고 있다. [3] 연구에서는 이미지 기반의 접근 방식으로 CNN(convolutional neural network)을 구성하여 EEG 신호를 분류하였다. LSTM을 사용하여 EEG 데이터를 분류한 [4]에서는 분류의 정확도를 높이기 위해, channel weighting 기법과 one dimension-aggregate approximation(1d-AX)을 사용하여 LSTM으로부터 효과적인 신호 표현을 추출해냈다. 연구 [2]에서는 raw 데이터를 Short-time Fourier Transform(STFT)로 변환한 후, batch normalization와 data augmentation을 포함하는 multi-input convolutional neural network을 사용하여 EEG 데이터를 분류하였다.

그러나, 기존 연구들은 각 사람의 동작을 분류하는 정확도는 높으나, 여러 사람의 공통 동작에 대한 분류 정확도는 각각에 비해 높지 않다는 한계점을 보이고 있다. 이는 EEG 신호가 사람마다 고유한 패턴을 가지고 있기 때문이다.[5]

이 논문에서는 기존 연구의 한계점을 개선하여, 여러 사람의 공통된 동작을 기존 연구보다 정확하게 분류할 수 있는 SSD+FBCSP+1D-Augmentation 기법을 제안한다. 이 논문에서 제안한 SSD+FBCSP+1D-Augmentation 기법은 Spatio-Spectral Decomposition(SSD) 필터와 Filter-Bank Common Spatial Patter(FBCSP) 필터를 기반으로 하고 있다. SSD는 불필요한 신호를 최소화하고, 원하는 신호 출력을 최대화하여 잡음을 줄이고, 특징적인 부분을 추출할 수 있으며, FBCSP는 개인의 EEG간 편차를 줄이기 위한 필터로, 다양한 주파수 밴드를 종합적으로 사용하여, 각각의 밴드에 Common Spatial Pattern(CSP)를 적용하였다. CSP 적용 후에는 Mutual Information Based Individual Feature(MIBIF)를 사용하여 추출된 특징을 선택적으로 사용하였다. 또한 윈도우 슬라이딩 기법을 기반으로 SSD와 FBCSP 필터를 통과한 데이터는 Long Short-Term Memory(LSTM)을 사용하여 분류하였다.

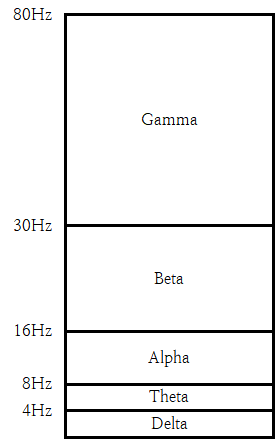
데이터의 수를 늘리는 방법으로는 Overlapping과 1D -Augmentation이 사용되었다. 일반적인 Augmentation은 이미지를 변형하여 데이터를 증가시키는 방법으로, 이미지가 2차원 데이터기 때문에 숫자 즉, 1차원 데이터에 대한 Augmentation 방법과는 차이가 있다. 따라서 본 연구에서는 숫자 데이터를 사용했기 때문에 1D – Augmentation을 통해 데이터 수를 증가시켰다.

이 논문은 다음과 같이 구성되었다. 2장에서는 이 연구와 밀접하게 관련된 연구들에 대해 기술하며, 3장에서는 이 논문에서 제안한 SSD+FBCSP+1D-Augmentation 기법을 상세히 기술한다. 그리고, 4장에서는 실험 결과를 기술하고, 5장에서는 결론 및 향후 계획에 대하여 기술한다.

2. 관련 연구

의학적으로 뇌파 데이터는 델타파(1~4Hz), 세타파(4~8Hz), 알파파(8~15Hz), 베타파(16~31Hz), 감마파(30~80Hz) 등의 여러 하위 대역으로 나눠진다[6]. 이 중에서 뇌파 분석에 보편적으로 사용되는 주파수는 알파파와 베타파에 해당하는 신호이다. 이완된 상태 또는 눈을 감은 상태에서 알파파 활성이 증가하고, 눈을 뜬 상태로 사고와 계산과 같은 정상적인 뇌활동을 하는 경우에는 베타파가 지배적이기 때문이다. 눈을 감으면 급격하게 베타파에서 알파파로 신호 패턴이 바뀌는 것을 통해 눈 움직임이 신호 변화에 큰 영향을 끼친다는 것을 알 수 있다[7]. BCI에서는 신호를 주파수 별로 분리하는 방법이 주로 사용된다. 각 주파수 별로 분리한 EEG 데이터에는 신체에서 발생되는 눈동자의 움직임에 의한 EOG 신호, 근육에 의한 EMG 신호, 심장 박동에 의한 ECG 신호 등에 의한 노이즈가 섞여 있기 때문에, 전처리 작업이 필수적이다[7].

[그림 1] 뇌파 데이터의 하위 대역



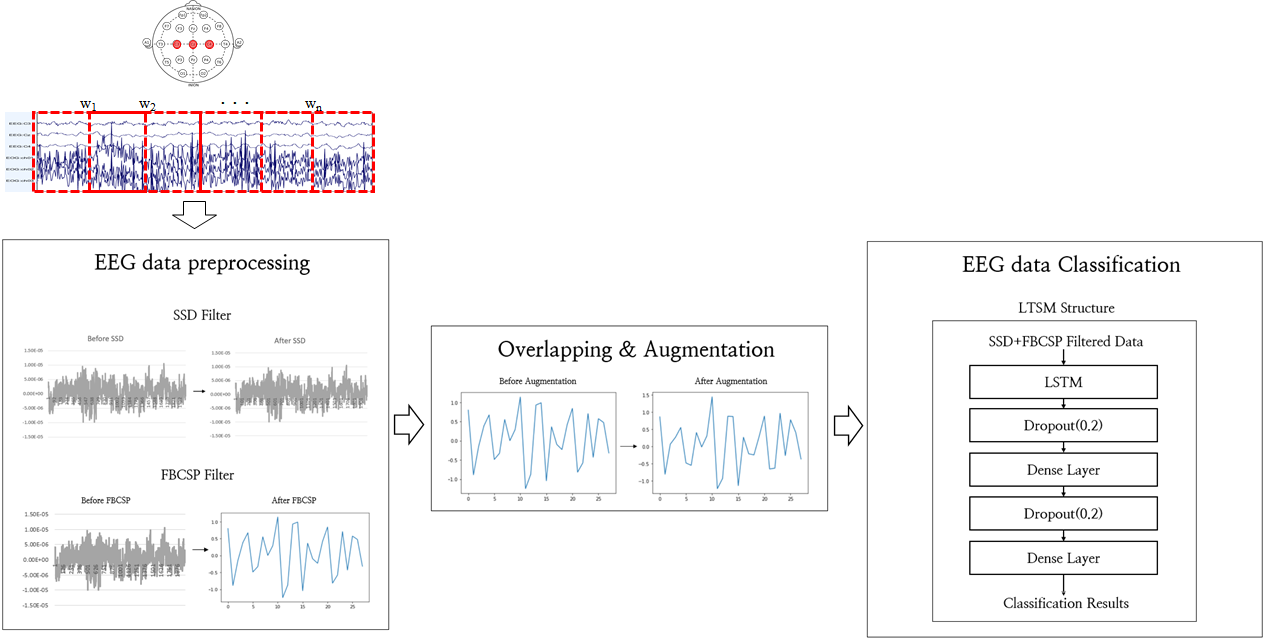
EEG 데이터 전처리에 Power Spectral Density(PSD) [8], Morlet wavelet, Short-time Fourier Transform(STFT)[9], Common Spatial Pattern(CSP), Spatio-Spectral Decomposition (SSD) 등의 알고리즘들이 사용되고 있다. 동작 상상 신호 전처리와 관련하여, [9], [10] 연구에서는 STFT를 이용해 동작 상상 신호(Motor Imagery Signal)를 이미지로 변형하여 사용하였다. CSP는 두 클래스 사이의 분산을 극대화하여 특징을 추출하는 방법으로 이미 많은 뇌파 분석에서 사용되고 있다[16]. 하지만 CSP는 데이터와 사람에 따라 편차가 있고, 이는 동작 주파수 대역별에 따라 CSP 필터의 성능이 달라진다는 것을 의미한다. 따라서 CSP 알고리즘에서는 넓은 주파수 대역 범위를 선택하거나, subject별 특정 주파수 범위를 수동적으로 선택하는 방법이 사용된다. 수동적으로 주파수 대역을 선택하는 번거로운 문제를 해결하기 위해 EEG의 주요한 특징들을 자율적으로 선택하여 수행하는 FBCSP가 고안되었다[9]. [12]에서는 여러 전처리 기법들을 가지고 실험한 결과, SSD와 CSP를 결합했을 때 정확도가 가장 높았다는 결론을 보여주고 있다. 하지만 위에 서술한 바와 같이 CSP는 각 사람에 대한 정확도는 높지만 여러 사람의 데이터를 가지고 동작을 분류할 때의 정확도는 낮다는 한계점을 가지고 있다. 따라서 이 논문에서는 SSD에 CSP가 아닌 FBCSP를 결합하여 사용하였다.

또한 전처리된 EEG를 분류하는 알고리즘으로 Support Vector Machines(SVM), Linear Discriminant Analysis(LDA), Naive Bayes(NB)등이 있다[22]. 최근에는 다양한 BCI 성능을 향상시키기 위한 CNN 기반의 모델들이 등장하고 있다. 그러나, CNN의 분류 정확도는 데이터가 왜곡될 때 낮아지고, 각 클래스 간 공간적 관계를 고려하는 것에 한계가 있다. 특히, EEG데이터는 같은 사람에서 측정되더라도 연속적이지 않고 왜곡이 심한 데이터 중 하나이다. 이러한 한계를 극복하기 위해 EEG 신호의 다양한 특성을 학습할 수 있는 CapsNet 그리고 EEG기반 BCI를 목적으로 만들어진 소형 CNN 구조의 EEGNet[23]과 같이 기존 CNN방식보다 더 좋고 강력한 모델이 개발되고 있다[9]. [10]에서는 원본 EEG 신호에 STFT를 적용하여 얻은 2차원 이미지를 ResNet과 multi-input CNN을 통해 분류하였다.

다양한 전처리 기법이나 분류 모델들이 제안되고 있지만, 여러 사람들의 EEG로부터 공통된 행동의 특징을 추출하여 분류하기에는 한계가 있다. 이 논문에서는 여러 사람의 동작 상상 뇌파로부터 동작을 분류하기 위하여, 여러가지 실험을 통해 선정된 SSD와 FBCSP 필터를 적용하여 LSTM으로 분류하는 SSD+FBCSP+1D-Augmentation 기법을 제안한다.

3. SSD+FBCSP+1D-Augmentation

[그림2] 제안된 시스템의 다이어그램



3.1 SSD

신경세포의 진동은 뇌의 다양한 인지, 지각 및 운동의 기초가 된다. 하지만 이러한 진동에 대한 EEG/MEG 기록은 매우 어려운 과정이다. EEG 신호는 기본적으로 많은 노이즈가 포함되어 있고, 다양한 진동이 복합적으로 중첩되어 있는 상태이기 때문이다. [24]에서는 다중 채널 EEG/MEG/LFP 기록으로부터 뉴런의 진동을 정확하고 빠르게 추출할 수 있는 방법을 제시한다.

[14]에서 제시한 방법은 데이터를 선형 분해하는 것에 기반을 둔다. 이는 피크 주파수(peak frequency)에서 신호 출력을 극대화하는 동시에 인접 주파수 bin에서 신호 출력을 최소화하여 신호 대 노이즈 비율을 최적화한다. 또한, 진동 관련 처리 과정에서 일반적으로 사용되는 방법과 동일하게 특징적인 스펙트럼 정보를 추출할 수 있다. 이 방법을 Spatio-spectral decomposition(SSD)이라고 부른다.

[14]에서 제안한 방법은 신호 대 잡음 비율이 1:10에 불과한 경우에도 진동 신호를 추출할 수 있을 정도로 우수한 성능을 가지고 있으며, 일반적으로 몇 분이 소요되는 다른 추출 기법과 달리 몇 밀리초 만에 실행이 된다는 장점을 가지고 있다.

EEG 신호 분석에 사용할 부분은 알파파(8~15Hz), 베타파(16~31Hz) 그리고 델타파(1~4Hz) 범위에 해당한다. 따라서 SSD에서 처리할 전체 주파수 범위는 0.2~30Hz, 제거할 white noise 범위는 10~12Hz, band stop filtering 처리를 할 부분은 9~13Hz로 변경하여 진행하였다.

3.2 CSP, FBCSP

Common Spatial Pattern(CSP), Filter Bank Common Spatial Pattern(FBCSP)는 EEG 분류에서 주로 사용되는 filter이다. Spatial filtering은 대뇌피질에서 운동 영역에 해당하는 뇌파 신호 변화 감지에 중요한 역할을 한다. CSP 알고리즘은 EEG 신호 특징 추출과 분석에서 주로 사용하는 다차원적 통계학의 일종으로, 서로 다른 두 클래스 사이의 평균 분산 비율을 최대화하는 최적의 공간 필터(Spatial filters)를 찾는다[16]. 공간 필터(Spatial filters)를 설계하기 위해 사용하는 방법은 두 공분산 행렬의 동시 대각화를 기반으로 한다[7]. CSP를 통하여 BCI의 성능을 크게 향상할 수 있다. CSP에 대한 이해를 위해, 각 N개의 채널로부터 T개의 샘플을 얻었다고 가정하였다. 이를 사용하여 1개의 trial에 대한 EEG 데이터를 N\*T 형태의 행렬 E로 나타낼 수 있다[7]. EEG의 normalized spatial covariance는 식 (1)과 같이 계산한다.

|  |  |
| --- | --- |
|  | (1) |

E'에서 첨자는 transpose 연산자를 나타내고, trace(𝑥)는 𝑥의 대각원소 합을 의미한다. 위의 과정을 통해 얻은 공분산 는 N\*N 형태로 표현할 수 있다. 분리할 두 클래스(왼쪽과 오른쪽 동작 상상)에 대한 공분산을 계산한 후 각 클래스 별로 공분산의 평균을 구함으로써 spatial covariance 을 구할 수 있다. 그 다음, 평균 공분산 행렬을 더하여 합성 공분산 행렬 를 구한다.

|  |  |
| --- | --- |
|  | (2) |

를 고유값 분해하여 얻은 고유벡터 와 고유값 를 사용하여 아래와 같이 백색화 변환 행렬(whitening transformation) 를 구할 수 있다.

|  |  |
| --- | --- |
|  | (3) |

백색화 변환 행렬은 각 클래스의 평균 공분산 행렬이 동일한 고유벡터를 가지도록 해준다. 백색화 변환 행렬은 다음과 같이 적용이 가능하다.

|  |  |
| --- | --- |
|  | (4) |
|  | (5) |

식 (4), (5)를 통하여 구한 , 는 동일한 고유벡터를 갖게 된다.

|  |  |
| --- | --- |
|  | (6) |

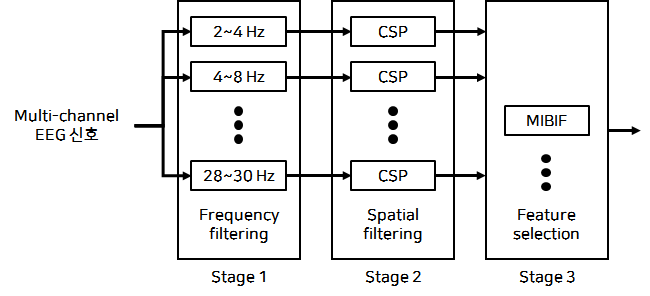
두 개의 고유값의 합은 항상 1이므로 에 대한 고유값이 가장 큰 고유벡터는 에 대해 가장 작은 고유값을 갖는다. 반대로 에 대해 고유값이 가장 큰 고유벡터는 에 대해 가장 작은 고유값을 갖는다. 이러한 특징은 고유벡터 가 두 분산을 분류하는데 유용하도록 만들어준다. 마지막으로 를 사용하여 CSP filter W를 계산하고, 식(7)을 통하여 공간 필터를 적용한 EEG 신호를 얻을 수 있다.

|  |  |
| --- | --- |
| *Z = WE* | (7) |

하지만 EEG 주파수 대역이 피실험자마다 차이가 있기 때문에 분류 가능한 주파수 대역이 CSP 알고리즘에 사용되지 않은 경우 분류 성능이 저하된다[17]. 이러한CSP의 단점을 보완하기 위해 FBCSP를 사용하였다. 기존 CSP와 달리 EEG 신호에서 중요한 부분을 자동적으로 선택하는 FBCSP에서는 EEG 신호 측정값이 다중 주파수 대역으로 bandpass-filtering된 후, 각 대역에서 CSP 특징들이 추출(feature extraction)된다. 특징 선택(feature selection) 알고리즘은 주파수 대역과 상응하는 CSP features로 이루어진 특징적인 한 쌍을 자동적으로 선택해낸다. 그 다음 분류 알고리즘을 사용하여 CSP features를 분류한다[9].

FBCSP는 크게 frequency filtering, spatial filtering, feature selection, classification 4개의 단계로 이루어진다. 첫 번째 단계에서 EEG 데이터는 여러 개의 주파수 대역으로 bandpass-filtering 된다. 그 다음 CSP 알고리즘을 사용하여 각 주파수 대역에서 spatial filtering을 수행한다. 여기에서 구한 bandpass와 spatial filter로 이루어진 쌍들은 각각 bandpass filter 주파수 대역 특유의 CSP features를 산출한다. 세 번째 단계에서는 feature selection 알고리즘이 filter bank로부터 차별적인 CSP features를 선택한다. 마지막 단계로, 분류 알고리즘이 선택된 CSP features를 분류한다[18].

[그림3] 논문에서 사용한 FBCSP 흐름도



3.3 Augmentation

CNN은 많은 데이터 분류에 사용되지만, 일반적으로 훈련을 위해 대규모 데이터셋을 필요로 한다. 데이터 증가(Augmentation)는 기존 데이터를 변환하여 새로운 데이터를 생성함으로써 제한된 데이터를 활용하는 방법이다[19]. 이러한 데이터 증가 방법은 학습 데이터의 크기를 늘리고, 학습 과정의 overfitting을 줄이는 것에 효과적이다.

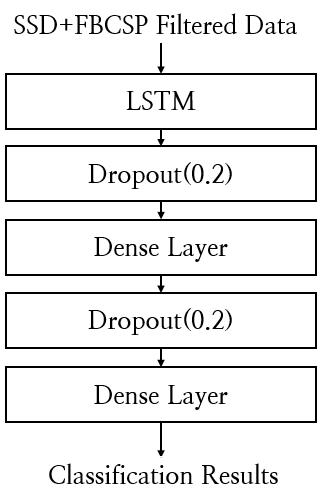
데이터 Augmentation의 관건은 정확한 라벨을 유지한 상태로 새 데이터를 생성하는 것이다. 일반적으로 Augmentation은 기존 데이터에서 정보를 변형하고 압축하여 기존 데이터와 동일한 라벨을 가지고 있는 새로운 데이터를 생성한다. Augmentation을 하는 방법에는 데이터 회전, 색상 변화, 데이터의 정체, 노이즈 등 여러 가지 방법들이 있다. [19]에서 Augmentation 적용 전에 비하여 적용 후의 정확도가 향상된 것을 확인할 수 있다.

일반적으로 Augmentation은2D 데이터의 증가에 사용하지만, 이 실험에서는 1D데이터를 증가시키기 위해 [19]에서 사용한 Jitter와 Scaling을 사용하였다. Jitter는 신호에 추가적인 노이즈를 추가하는 방법으로, 노이즈에 데이터가 영향을 받는 것을 감소시킴으로써 성능을 향상시키는 것에 도움이 된다. Scaling은 랜덤 스칼라를 곱하여 윈도우 내의 데이터 규모를 늘리는 방법이다.

3.4 LSTM

Recurrent Neural Networks(RNN)는 RNN 모듈을 사용하여 얻은 이전 상태 값을 다음 작업에서 적용할 수 있다는 장점을 가지고 있다.  하지만 RNN은 back-propagation방식을 사용해서 학습을 하기 때문에 gradient 정보가 갈수록 소실된다는 한계점을 가지고 있다. 이 문제를 해결하기 위해, Long Short-Term Memory Network (LSTM)이 사용되었다. LSTM은 과거의 셀 상태 정보를 은닉층의 연산과 함께 다음 은닉층으로 전달한다. LSTM은 크게 3가지의 게이트로 정보의 사용 유무를 판단한다.

[그림4] 사용한 LSTM구조



첫 번째 게이트는 forget게이트로, 벡터 sigmoid를 이용하여 삭제할 정보를 결정한다. 는 입력 벡터이다.

|  |  |
| --- | --- |
|  | (8) |

두 번째 층은 입력 게이트 벡터로, 새로운 정보를 획득하는 가중치 역할을 한다. 어떤 값이 갱신될 것인지 결정하는sigmoid층과 갱신된 값의 벡터를 생성하는tanh층으로 이루어져 있다.

|  |  |
| --- | --- |
|  | (9) |
|  | (10) |

셀의 상태는 (4)번 식을 거쳐, 이전 셀의 상태로부터 새로운 셀의 상태가 결정된다.

|  |  |
| --- | --- |
|  | (11) |

마지막은 출력 게이트이다. 은닉층의 연산 결과 값인 는 (5), (6)번 식을 통해 계산된다.

|  |  |
| --- | --- |
|  | (12) |
|  | (13) |

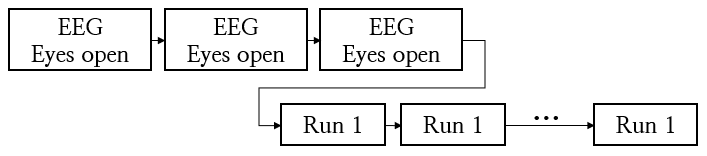
4. 실험결과

4.1 데이터셋과 실험 환경

왼손, 오른손을 구분하기 위한 데이터로EEG 신호 BCI competition IV 2b dataset[13]을 사용하였다. 해당 데이터셋은 모니터에 나타난 화살표 방향을 보고, 피실험자가 동일한 방향을 생각하도록 지시하여 수집한 데이터들로 이루어져 있다. 데이터 수집은 9명의 subject에 대해 진행되었다. 데이터 수집 방법은 다음과 같다.

각 subject는 의자에 앉아, 눈높이에서 1m 떨어진 장소에서 표시되는 모니터를 응시한다.  각 세션의 시작에서 subject는 약 5분의 시간이 주어진다. 데이터 측정은 크게 3단계로 나누어진다. (1) 눈을 뜬 상태로 2초 동안 화면의 정지된 십자 표시 응시한 후, (2) 눈을 감은 상태로 1초, (3) 2초 동안 눈을 움직이는 단계로 이루어져 있다.

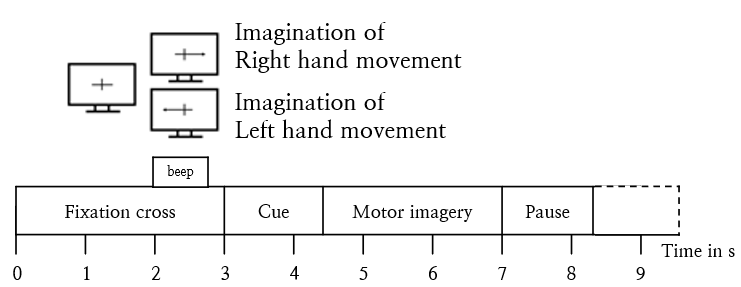
[그림5] 각 세션의 구성



각각의 trial은 고정된 십자와 짧은 기계음(1kHz, 70ms)으로 시작한다. 몇 초 후에 왼쪽이나 오른쪽 화살표가 1.25s 동안 화면에 표시된다. 그 다음, 각 subject는 4s 동안 화살표에 해당하는 손의 움직임을 상상한다. 각 trial의 마지막은 1.5s의 휴식기간으로 이루어진다. 이번 trial의 영향이 다음 trial에 반영되지 않도록, 1s 이내의 시간이 추가로 주어진다.

실험은 3개의 양극 전극(C3, Cz, C4)을 사용하여 250hz의 샘플링 주파수로 기록된다. 기록된 신호는 0.5Hz~100Hz에서 bandpass -filtering되었고, 50Hz에서 Notch filter가 적용되었다. 데이터 셋에서는 각 subject 당 5개의 세션 데이터를 제공하는데, 본 논문에서는 첫번쨰와 두번째 세션 데이터 즉, feedback이 없는 데이터를 사용하였다. 5개 중 첫번쨰와 두번째 세션 데이터는 feedback이 없고, 세번째, 네번째, 그리고 다섯번 째에서는 smiley feedback이 주어진다.

[그림6] 각 trial의 구성

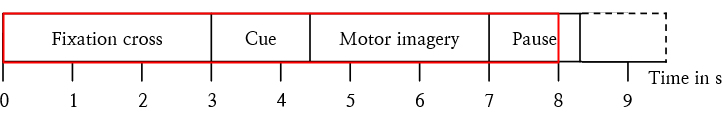


4.2 데이터 추출 범위 변경

BCI competition IV 2b 데이터 셋에서 가져온 데이터는 뇌파 신호 값이 연속적으로 나열되어 있다. 이 데이터에는 EEG 외에 EOG 신호가 합쳐져 있는 상태이기 때문에 전처리를 통해 EOG를 제거했다. 또한, 기존 데이터 셋에서 rejected trial로 표시된 trial들은 이 연구에서 왼손과 오른손을 구분하는데 부적합한 데이터라고 판단되어 제거한 후 진행하였다. Trial 별로 데이터를 classification하기 위해서는 이 데이터를 1개의 trial 시간으로 나누어야 한다. 하지만 [13]에서와 [그림 5]에서 1개 trial 측정이 끝나는 시간은 8~9s로 trial마다 다른 것을 확인할 수 있다. 좀 더 정확한 왼쪽, 오른쪽 방향 구분 판단을 위한 1개의 trial 시간을 확인하기 위해서 여러가지 범위로 실험을 진행하였다.

실험은 세가지 방식으로 진행되었다. 첫 번째는 1개 trial 길이를 9초(2250Hz)로 정한 후 해당 trial에서 4~7s만 데이터를 가져오는 방식이다([그림 4]에서 motor imagery에 해당하는 부분). 이 데이터에 FBCSP 필터를 적용한 후 LSTM을 실행한 결과, 전체 subject에 대해 0.71의 정확도를 얻을 수 있었다. 두 번째 시도는 위와 동일하게 1개 trial 길이를 9s로 정했지만, 실제 사용할 범위를 cue data까지 포함한 3~7s로 변경하였다. 그 결과 FBCSP 적용 후 LSTM에 대한 정확도로 0.76, 즉 더 향상된 결과를 얻을 수 있었다. 앞의 두 경우는 pause가 끝난 후의 데이터도 포함하고 있기 때문에 불필요한 데이터를 줄이고자 pause가 끝나기 전인 8s(2000Hz)를 1개 trial 길이로 정하였다. 또한 이전과 달리 cue가 시작되기 전 데이터까지 포함하기 위해 추출할 범위를 0~6s로 변경하였다. 이에 대한 LSTM 결과는 0.77로 첫 번째, 두 번째 경우보다 소폭 상승하였다.

[그림7] 1개 trial 구성



4.3 실험 결과

본 실험에서 데이터 셋으로 사용한 BCI competition IV 2b dataset의 피실험자는 총 9명으로, 각 피실험자 별 오른쪽과 왼쪽 trial의 개수에 차이가 있다. 전처리를 하기 전에 매트랩을 사용하여 rejected trial을 제거하였다. 한 trial당 최소 8초 최대 9초의 길이를 갖기 때문에 2000Hz단위로 연속적인 raw EEG 데이터를 나누었다. 이렇게 나누어진 데이터는 1개의 trial 당 0~8초의 길이를 갖게 된다.

[표 1] subject 별 trial 개수

|  |  |  |
| --- | --- | --- |
|  | left | right |
| sub1 | 59 | 62 |
| sub2 | 61 | 62 |
| sub3 | 55 | 56 |
| sub4 | 75 | 74 |
| sub5 | 72 | 71 |
| sub6 | 53 | 57 |
| sub7 | 67 | 67 |
| sub8 | 65 | 57 |
| sub9 | 59 | 60 |

이후 한 개의 trial마다 SSD를 처리하였다. [14]에서는 Alpha파 대역에만 SSD를 적용하였다. 일반적으로 뇌파 분류에는 Alpha파와 Beta파 대역이 사용된다. 하지만 [2]에서 Delta파 대역을 포함하여 EEG 데이터 분류를 실행한 결과 약 6.96% 정도 정확도가 증가한다는 결과를 확인할 수 있다. 따라서 본 연구에서는 Alpha(8~14Hz)파 대역에 SSD를 적용한 경우, Alpha파와 Beta파 대역에 SSD를 적용한 경우 그리고 Alpha파, Beta파, Delta파 대역에 SSD를 적용한 경우를 비교하였다.

Alpha파 대역에만 SSD를 적용한 경우의 데이터 처리는 다음과 같다. 일반적인 Alpha파 범위는 8~14Hz로, 이 부분에 band-pass filter을 적용한다. White noise는 제거해야 하는 범위로 10~12Hz에 해당한다. White noise가 영향을 줄 수 있는 인접한 범위인 9~14Hz까지 band-stop filtering을 수행한다. SSD를 적용한 후에는 CSP 또는 FBCSP를 적용하였다.

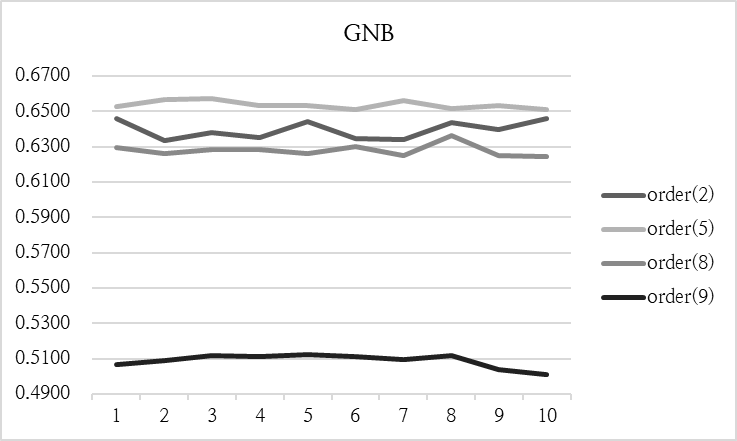
다음으로는 실제로 CSP에 비해 FBCSP가 더 높은 정확도를 보이는지 확인하기 위해 비교 실험을 진행하였다. [표 3], [표 4]는 CSP와 FBCSP 각 알고리즘에서 가장 좋은 정확도를 보인 윈도우 크기와 order값에서의 결과이다. CSP는 윈도우 크기 600Hz, FBCSP는 윈도우 크기 700Hz로 실행하였다. CSP 알고리즘에서는 butter-worth 필터가 사용되었다. 여기서 butter-worth의 order값을 변경함에 따라 분류 정확도가 달라지는 것을 확인하였다. 기존 주파수 범위 7~30Hz에서 8~14H로 변경 후 order 값 1, 2, 5, 7, 8, 9에 대해 결과값을 비교하였다. 3개의 알고리즘 GNB, SVM, LDA 중에서 GNB가 가장 분류 정확도가 낮았기 때문에 나머지 2개 알고리즘 SVM, LDA를 기준으로 정확도가 높게 나온 order값을 선택하였다.

[표 2] 전체 subject에 대한butter-worth order 값 변경 결과

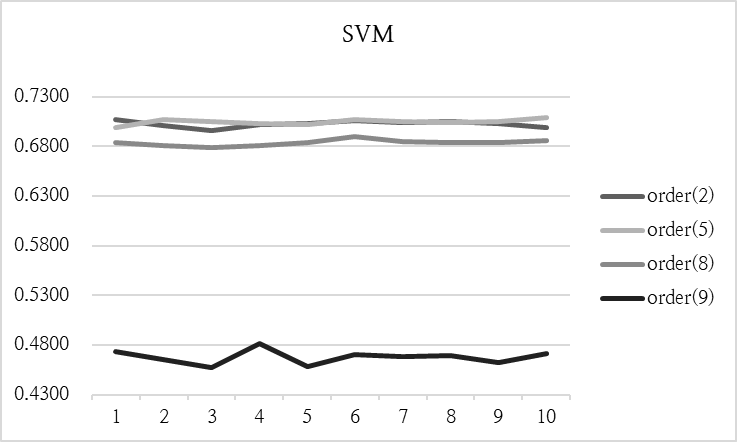
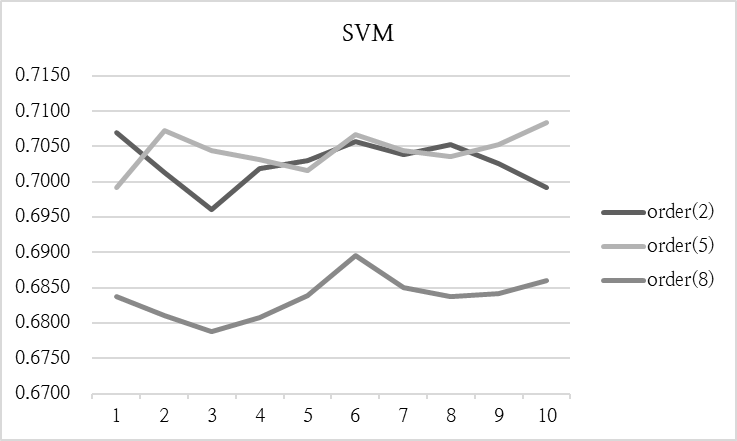
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| order  (회) | 2 | | | 5 (기존) | | |
| GNB | SVM | LDA | GNB | SVM | LDA |
| 1 | 0.6461 | 0.707 | 0.7147 | 0.6525 | 0.6992 | 0.7031 |
| 2 | 0.6335 | 0.7013 | 0.7117 | 0.6567 | 0.7073 | 0.7083 |
| 3 | 0.6378 | 0.6961 | 0.7068 | 0.6573 | 0.7044 | 0.7065 |
| 4 | 0.6349 | 0.7019 | 0.7124 | 0.653 | 0.7032 | 0.7122 |
| 5 | 0.6442 | 0.703 | 0.712 | 0.6535 | 0.7016 | 0.707 |
| 6 | 0.6347 | 0.7057 | 0.7085 | 0.6509 | 0.7067 | 0.7101 |
| 7 | 0.6339 | 0.7038 | 0.7067 | 0.6563 | 0.7044 | 0.7112 |
| 8 | 0.6435 | 0.7052 | 0.7112 | 0.6515 | 0.7035 | 0.7065 |
| 9 | 0.6396 | 0.7026 | 0.7085 | 0.6532 | 0.7052 | 0.706 |
| 10 | 0.6458 | 0.6992 | 0.7087 | 0.6509 | 0.7084 | 0.7062 |
| 평균 | 0.6394 | 0.70258 | 0.71012 | 0.65358 | 0.70439 | 0.70771 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 8 | | | 9 | | |
| GNB | SVM | LDA | GNB | SVM | LDA |
| 0.6292 | 0.6837 | 0.6888 | 0.5068 | 0.4731 | 0.4884 |
| 0.6263 | 0.681 | 0.6937 | 0.5091 | 0.4656 | 0.4802 |
| 0.6286 | 0.6788 | 0.6899 | 0.5116 | 0.4571 | 0.468 |
| 0.6284 | 0.6807 | 0.6883 | 0.511 | 0.4814 | 0.4888 |
| 0.6263 | 0.6839 | 0.6881 | 0.5122 | 0.4581 | 0.4754 |
| 0.6301 | 0.6896 | 0.6876 | 0.511 | 0.4705 | 0.4799 |
| 0.625 | 0.685 | 0.6856 | 0.5092 | 0.4687 | 0.4724 |
| 0.6363 | 0.6838 | 0.6931 | 0.5115 | 0.4695 | 0.4757 |
| 0.6252 | 0.6841 | 0.6915 | 0.504 | 0.4626 | 0.4746 |
| 0.6242 | 0.686 | 0.6873 | 0.5011 | 0.4717 | 0.4799 |
| 0.62796 | 0.68366 | 0.68939 | 0.50875 | 0.46783 | 0.47833 |

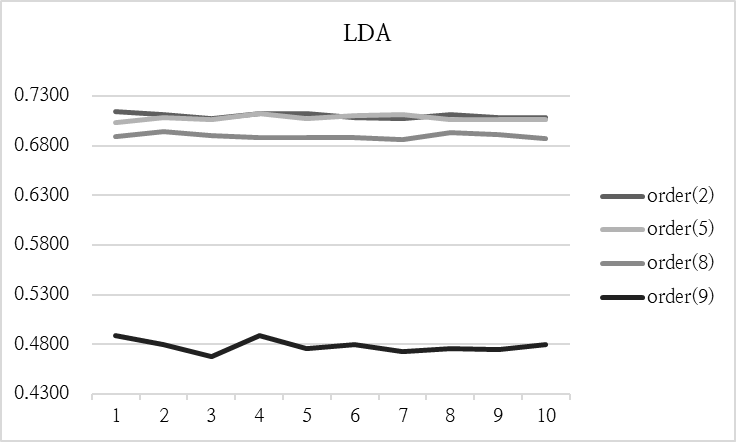
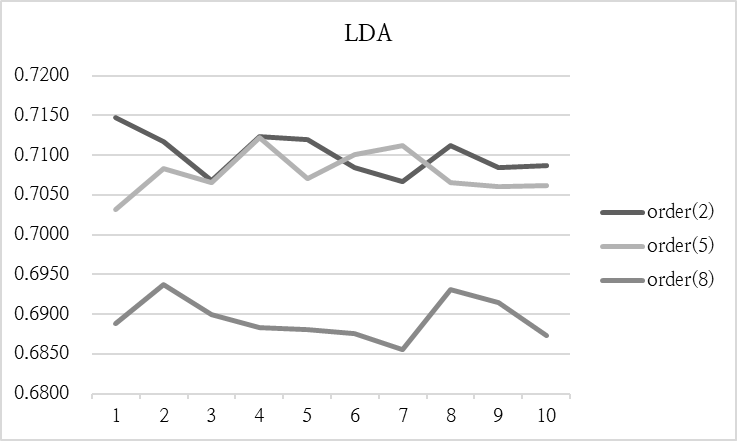
[그림 8] GNB 분류 알고리즘 사용 시 order 별 결과값 비교 그래프



[그림 9] SVM 분류 알고리즘 사용 시 order 별 결과값 비교 그래프

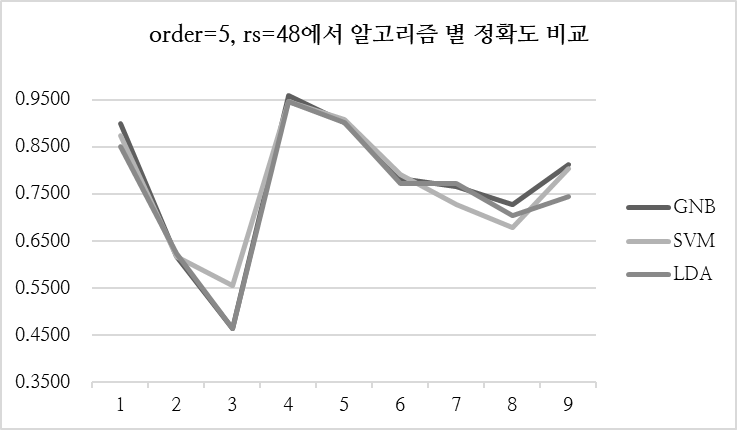
 

[그림 10] LDA분류 알고리즘 사용 시 order 별 결과값 비교 그래프

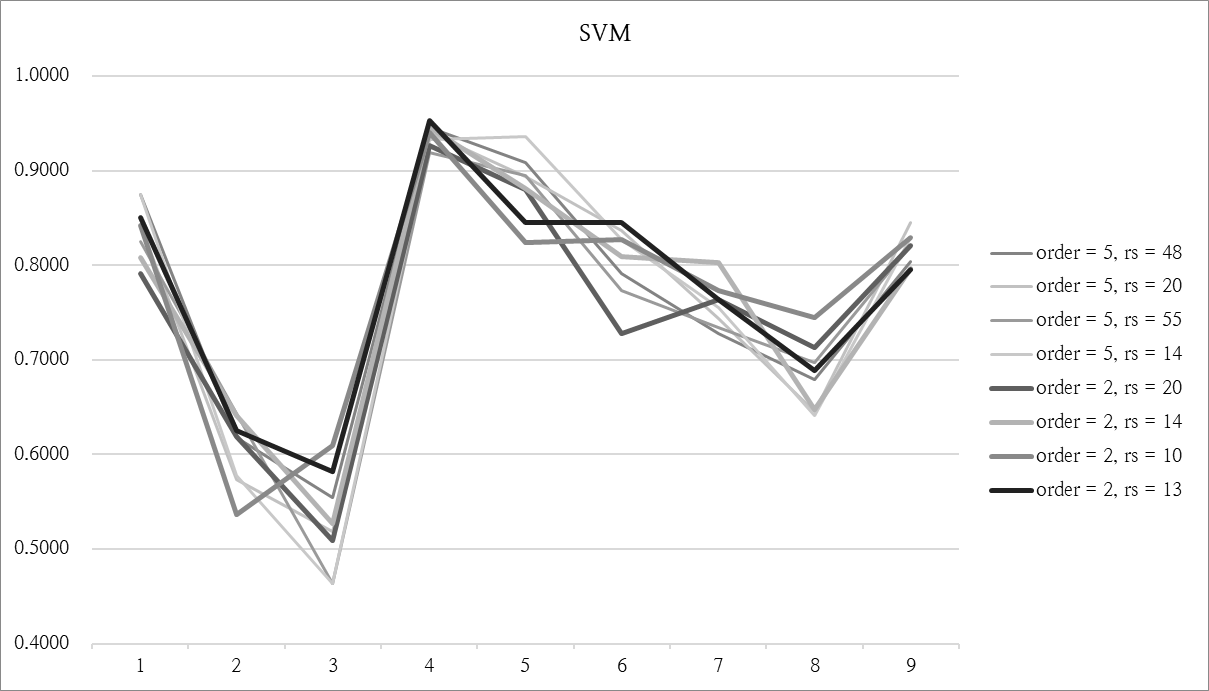
 

위 표와 그래프를 보면, 전체 subject에 CSP를 적용한 후 분류하였을 때 order 값이 2 또는 5인 경우 가장 좋은 결과를 얻을 수 있었다. 위 결과에 따라 CSP 알고리즘 적용 시, butter-worth 필터에서는 order를 기존 값 5에서 2로 변경하여 실행하였다. FBCSP 알고리즘에서 사용되는 Chebyshev 필터에서도 동일한 분류 알고리즘(GNB, SVM, LDA)으로 정확도를 분석하였다. FBCSP에서는 알고리즘 별로 비교한 그래프[그림 8], [그림 9], [그림 10]을 보았을 때 GNB, SVM, LDA 중 SVM이 subject별 차이가 가장 적었기 때문에 SVM을 기준으로 order, rs 값을 선택하였다. 전체 subject에 대하여 SVM 기준으로 정확도를 비교하였을 때, order = 2, rs = 13에서 subject별 차이가 가장 적은 것을 확인할 수 있다[그림 12], [표 3].

[그림 11] order=5, rs=48 기존 값에서 분류 알고리즘 별 정확도 비교



[그림 12] SVM 기준 order, rs 별 정확도 비교



[표 3] 전체 subject에 대한 Chebyshev order, rs 값 변경 결과

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| order sub | order = 5, rs = 48 (기존) | | | order = 5, rs = 20 | | |
| GNB | SVM | LDA | GNB | SVM | LDA |
| 1 | 0.9000 | 0.8750 | 0.8500 | 0.8667 | 0.8417 | 0.8667 |
| 2 | 0.6167 | 0.6173 | 0.6244 | 0.6212 | 0.5731 | 0.6282 |
| 3 | 0.4636 | 0.5545 | 0.4636 | 0.5182 | 0.5182 | 0.5636 |
| 4 | 0.9600 | 0.9457 | 0.9467 | 0.9533 | 0.9400 | 0.9262 |
| 5 | 0.9010 | 0.9081 | 0.9010 | 0.8862 | 0.8933 | 0.8938 |
| 6 | 0.7818 | 0.7909 | 0.7727 | 0.8091 | 0.8264 | 0.7727 |
| 7 | 0.7654 | 0.7275 | 0.7725 | 0.8033 | 0.7434 | 0.7879 |
| 8 | 0.7282 | 0.6789 | 0.7032 | 0.7045 | 0.6455 | 0.7026 |
| 9 | 0.8129 | 0.8038 | 0.7455 | 0.8705 | 0.8455 | 0.7780 |
| 평균 | 0.7700 | 0.7669 | 0.7533 | 0.7814 | 0.7586 | 0.7689 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| order = 2, rs = 13 | | | order = 2, rs = 14 | | |
| GNB | SVM | LDA | GNB | SVM | LDA |
| 0.8667 | 0.8500 | 0.8333 | 0.8583 | 0.8083 | 0.8333 |
| 0.6340 | 0.6256 | 0.5763 | 0.6404 | 0.6210 | 0.6006 |
| 0.5182 | 0.5818 | 0.5545 | 0.5455 | 0.5273 | 0.5091 |
| 0.9600 | 0.9533 | 0.9529 | 0.9529 | 0.9457 | 0.9667 |
| 0.7676 | 0.8452 | 0.8938 | 0.7829 | 0.8814 | 0.8743 |
| 0.7909 | 0.8455 | 0.7727 | 0.8000 | 0.8091 | 0.8000 |
| 0.7341 | 0.7637 | 0.7709 | 0.7797 | 0.8033 | 0.8044 |
| 0.7205 | 0.6891 | 0.6885 | 0.6962 | 0.6474 | 0.6712 |
| 0.8379 | 0.7955 | 0.7697 | 0.8144 | 0.7962 | 0.7447 |
| 0.7589 | 0.7722 | 0.7570 | 0.7634 | 0.7600 | 0.7560 |

[표 4]과 [표 5]을 비교했을 때, 전체적으로 CSP에 비해 FBCSP의 정확도가 높은 것을 볼 수 있다.

[표 4] CSP 적용 후 classification 실행 결과(왼)

[표 5] FBCSP 적용 후 classification 실행 결과(오)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | GNB | SVM | LDA |  |  | GNB | SVM | LDA |
| sub1 | 0.57 | 0.62 | 0.64 |  | sub1 | 0.64 | 0.66 | 0.67 |
| sub2 | 0.56 | 0.55 | 0.6 |  | sub2 | 0.54 | 0.57 | 0.52 |
| sub3 | 0.5 | 0.52 | 0.57 |  | sub3 | 0.55 | 0.56 | 0.53 |
| sub4 | 0.82 | 0.86 | 0.87 |  | sub4 | 0.86 | 0.85 | 0.87 |
| sub5 | 0.82 | 0.86 | 0.87 |  | sub5 | 0.82 | 0.83 | 0.82 |
| sub6 | 0.59 | 0.71 | 0.73 |  | sub6 | 0.73 | 0.74 | 0.74 |
| sub7 | 0.48 | 0.6 | 0.58 |  | sub7 | 0.71 | 0.73 | 0.77 |
| sub8 | 0.65 | 0.63 | 0.64 |  | sub8 | 0.66 | 0.65 | 0.66 |
| sub9 | 0.6 | 0.72 | 0.74 |  | sub9 | 0.72 | 0.6 | 0.71 |
| 평균 | 0.62 | 0.67 | 0.69 |  | 평균 | 0.69 | 0.7 | 0.7 |

추가적으로 FBCSP가 실제로 전체 subject를 입력 데이터로 사용했을 때에도 정확도가 낮아지지 않게 하는데 효과적인지 확인하기 위해 SSD를 적용하지 않은 상태에서 raw EEG 데이터와 이 데이터에 FBCSP를 적용한 경우, STFT를 적용한 경우를 비교해보았다. [표 6]는 SSD를 적용하지 않은 raw EEG 데이터에 FBCSP를 적용한 후 GNB, SVM, LDA 알고리즘을 실행한 결과이고, [표 7]는 SSD를 적용하지 않은 raw EEG 데이터를 STFT를 사용하여 스펙트로그램으로 변환시킨 후 SVM, RF, KNN 알고리즘을 실행한 결과이다. [표 7]는 subject별 accuracy는 낮지만, 전체 subject와 각 subject 별 accuracy 차이가 적다. 반면, [표 7]는 subject 별 accuracy는 높지만, 전체 subject와 각subject 별로 실행했을 때 결과값 차이가 크다. SSD 처리를 후에 피실험자에 따른 주파수 대역 차이를 줄이기 위해 FBCSP이 효과적이라는 것을 알 수 있었고, 따라서 본 실험에서는 FBCSP를 적용하였다.

[표 x] raw EEG 데이터

|  |  |  |  |
| --- | --- | --- | --- |
|  | GNB | SVM | LDA |
| sub1 | 0.55 | 0.33 | 0.45 |
| sub2 | 0.48 | 0.39 | 0.58 |
| sub3 | 0.50 | 0.47 | 0.47 |
| sub4 | 0.90 | 0.63 | 0.50 |
| sub5 | 0.67 | 0.49 | 0.59 |
| sub6 | 0.57 | 0.50 | 0.60 |
| sub7 | 0.67 | 0.72 | 0.47 |
| sub8 | 0.73 | 0.73 | 0.55 |
| sub9 | 0.63 | 0.47 | 0.47 |
| 전체 sub | 0.66 | 0.61 | 0.58 |

[표 6] raw EEG 데이터 + FBCSP

|  |  |  |  |
| --- | --- | --- | --- |
|  | GNB | SVM | LDA |
| sub1 | 0.75 | 0.76 | 0.72 |
| sub2 | 0.50 | 0.48 | 0.51 |
| sub3 | 0.52 | 0.49 | 0.51 |
| sub4 | 0.89 | 0.89 | 0.89 |
| sub5 | 0.83 | 0.88 | 0.90 |
| sub6 | 0.77 | 0.73 | 0.71 |
| sub7 | 0.82 | 0.82 | 0.83 |
| sub8 | 0.71 | 0.66 | 0.71 |
| sub9 | 0.78 | 0.75 | 0.75 |
| 전체 sub | 0.73 | 0.72 | 0.72 |

[표 7] raw EEG 데이터 + STFT

|  |  |  |  |
| --- | --- | --- | --- |
|  | SVM | RF | KNN |
| sub1 | 0.85 | 0.85 | 0.91 |
| sub2 | 0.86 | 0.83 | 0.88 |
| sub3 | 0.86 | 0.89 | 0.88 |
| sub4 | 0.93 | 0.88 | 0.92 |
| sub5 | 0.89 | 0.83 | 0.90 |
| sub6 | 0.85 | 0.85 | 0.87 |
| sub7 | 0.87 | 0.84 | 0.89 |
| sub8 | 0.89 | 0.84 | 0.91 |
| sub9 | 0.90 | 0.79 | 0.83 |
| 전체 sub | 0.66 | 0.65 | 0.63 |

[표 4], [표 5]에 따라 CSP에 비해 FBCSP를 적용한 결과의 정확도가 더 높은 것을 확인하였고, [표 x]와 [표 6]를 비교하였을 때, raw EEG데이터에 FBCSP를 적용한 경우가 정확도가 더 높은 것을 확인할 수 있다. [표 6]와 [표 7]를 비교했을 때에는 raw EEG 데이터에 STFT로 스펙트로그램 변환을 하여 분류했을 때보다 FBCSP 필터를 적용하여 분류한 경우가 정확도의 결과가 더 높다. 그 다음으로는 데이터 전처리 방식으로 CSP, FBCSP, STFT 중에서 FBCSP를 사용한다고 가정했을 때, SSD를 사용한 경우와 사용하지 않은 경우를 비교하였다. [표 6]와 [표 8]을 비교한 결과 SSD 적용 전에 비하여, SSD를 적용했을 때의 정확도가 더 높은 것을 확인하였다.

[표 8] SSD(8~30Hz) + FBCSP

|  |  |  |  |
| --- | --- | --- | --- |
|  | GNB | SVM | LDA |
| sub1 | 0.78 | 0.74 | 0.74 |
| sub2 | 0.54 | 0.51 | 0.52 |
| sub3 | 0.58 | 0.61 | 0.57 |
| sub4 | 0.92 | 0.88 | 0.9 |
| sub5 | 0.84 | 0.89 | 0.91 |
| sub6 | 0.81 | 0.76 | 0.78 |
| sub7 | 0.87 | 0.84 | 0.86 |
| sub8 | 0.7 | 0.62 | 0.62 |
| sub9 | 0.81 | 0.78 | 0.75 |
| 전체 sub | 0.76 | 0.74 | 0.74 |

오버래핑과 Augmentation에 따른 정확도 향상 효과를 확인하기 위해 raw 데이터에 대해 STFT와 각각 다른 오버래핑 비율을 적용했다. 같은 비율의 Augmentation 상황에서 오버래핑 비율을 10~90%의 10단위로 적용하였다. 비교 결과, 오버래핑이 10%이였을 때의 정확도가 가장 높았다. 아래 [표 9]에서는 SSD 적용 데이터에 대한 오버래핑 비율에 따른 정확도를 비교하였다. [표 9]은 SSD 주파수 대역, CSP/FBCSP 적용 주파수 대역, 오버래핑 적용 비율 별로 전체 subject (1~9)에 대한 train acc와 test acc를 정리한 표이다.

[표 9] SSD, CSP/FBCSP 주파수 범위, 오버래핑 비율 별 정확도

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | | | 오버래핑 | train | test |
| SSD | 8~30 | CSP | 8~14 | 0% | 0.81 | 0.8 |
| 10% | 0.82 | 0.79 |
| 90% | 0.83 | 0.81 |
| FBCSP | 8~30 | 0% | 0.92 | 0.75 |
| 10% | 0.79 | 0.77 |
| 90% | 0.89 | 0.86 |
| 8~14 | CSP | 8~14 | 0% | 0.81 | 0.75 |
| 10% | 0.82 | 0.79 |
| 90% | 0.85 | 0.82 |
| FBCSP | 8~14 | 0% | 0.78 | 0.76 |
| 10% | 0.76 | 0.75 |
| 90% | 0.82 | 0.8 |

SSD 범위 8~30Hz, FBCSP 범위 8~30Hz, 오버래핑 비율 90%에서 accuracy가 가장 높게 나오는 것을 확인할 수 있다. 동일한 조건으로 각 subject에 대한 LSTM train/test 결과는 다음과 같다.

[표 10] subject 별 train/test 결과

|  |  |  |
| --- | --- | --- |
|  | train | test |
| sub1 | 0.96 / 0.17 | 0.93 / 0.17 |
| sub2 | 0.96 / 0.12 | 0.83 / 0.58 |
| sub3 | 0.89 / 0.24 | 0.75 / 0.61 |
| sub4 | 0.99 / 0.02 | 1 / 0.04 |
| sub5 | 0.99 / 0.05 | 0.95 / 0.23 |
| sub6 | 0.99 / 0.02 | 0.94 / 0.22 |
| sub7 | 0.94 / 0.13 | 0.88 / 0.35 |
| sub8 | 0.92 / 0.19 | 0.85 / 0.43 |
| sub9 | 0.97 / 0.25 | 0.92 / 0.25 |
| 전체 sub | 0.89 / 0.22 | 0.86 / 0.32 |

SSD와 FBCSP로 전처리를 끝낸 데이터에 1\_Dimensional Augmentation을 추가하여 데이터의 전체 양을 늘려준다. Augmentation은 DA\_Jitter와 DA\_Scaling을 사용하였다. 기존 데이터 11098개에 Jitter를 적용한 데이터 11098개, Scaling을 적용한 데이터 11098개가 추가되었다. Augmentation에 데이터를 증가시키는 함수가 많지만, 2개만 선택하여 사용한 이유는 1개의 Augmentation함수를 통해서도 높은 정확도 향상 효과를 얻을 수 있었기 때문이다. 실제로 subject2의 경우, Jitter의 적용만으로 왼손/오른손 분류 정확도가 82%에서 93%로 9% 증가한 것을 볼 수 있었다.

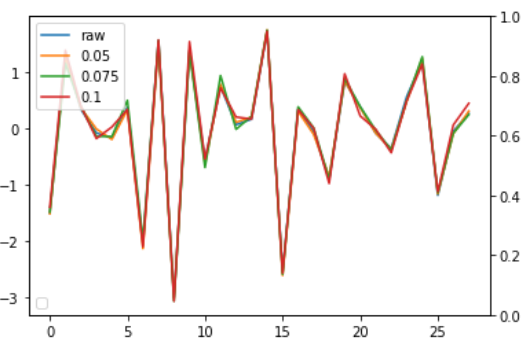
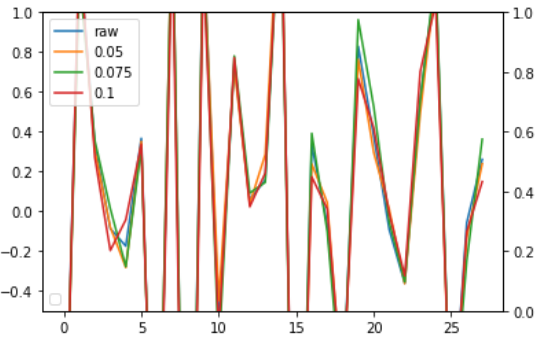
다음은 1D-Augmentation까지 처리한 데이터로 LSTM을 실행시킨 결과이다. Augmentation을 적용하기 전에 비하여 accuracy가 높아진 것을 확인할 수 있다. Augmentation 기존 인자를 그대로 적용한 결과는 [표 11]과 같다.

[표 11] Augmentation 적용 후 정확도

|  |  |  |
| --- | --- | --- |
|  | train | test |
| sub1 | 1 / 0 | 0.99 / 0.03 |
| sub2 | 1 / 0 | 1 / 0.02 |
| sub3 | 0.99 / 0.02 | 0.98 / 0.08 |
| sub4 | 1 / 0 | 1 / 0.01 |
| sub5 | 0.99 / 0.03 | 0.99 / 0.04 |
| sub6 | 1 / 0.01 | 1 / 0 |
| sub7 | 1 / 0 | 1 / 0.08 |
| sub8 | 1 / 0 | 0.99 / 0.09 |
| sub9 | 1 / 0 | 0.99 / 0.6 |

하지만, 결과를 보면raw 데이터에 대한 변경된 비율이 적어 중복의 발생으로 인해 결과값이 0.99~1.00으로 매우 높게 나온다. 따라서 데이터의 중복을 피하기 위해 augmentation을 적용하지 않은 raw 데이터와 augmentation을 각각 0.05, 0.075, 0.1로 변경하였을 때 결과값을 비교하여 sigma 값을 선택하였다[그림 13]. 결과적으로 sigma의 값 기존 0.05에서 0.075로 변경하여 데이터의 변화를 이전보다 크게 줌으로써 데이터의 양을 증가시키는 방법을 선택하였다([표 12]의 “Proposed method + Augmentation”).

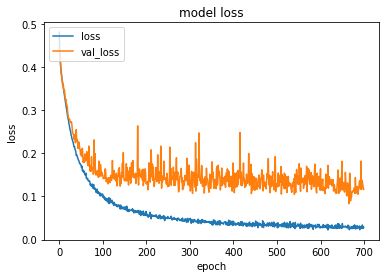
[그림 13] sigma 변경에 따른 원본 데이터와의 차이 비교

각 subject 별 accuracy가 높아진 것을 확인한 후 전체 subject에 대한 데이터에서도 accuracy를 확인하였다. 전체 subject에 대한 classification은 accuracy 0.972로 전체 피실험자에 대한 데이터를 입력하였음에도 각 subject 데이터를 입력했을 때와 별 차이 없이 높은 accuracy가 나왔음을 확인할 수 있다.

[그림 14] 전체 subject에 대한 classification 결과

스크린샷이(가) 표시된 사진

매우 높은 신뢰도로 생성된 설명

[표 12]는 다른 논문에서 다룬 알고리즘들에 대한 결과와 본 연구에서 제안한 알고리즘에 대한 결과를 정리한 표이다.

[표 12] 알고리즘 별 정확도 비교

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | DeepNet[9] | EEGNet[9] | CNN[9] | CapsNet[9] | CNN-SAE[25] |
| sub1 | 0.67 | 0.67 | 0.69 | 0.78 | 0.78 |
| sub2 | 0.56 | 0.58 | 0.54 | 0.55 | 0.63 |
| sub3 | 0.54 | 0.55 | 0.52 | 0.55 | 0.60 |
| sub4 | 0.94 | 0.95 | 0.95 | 0.95 | 0.95 |
| sub5 | 0.84 | 0.86 | 0.85 | 0.83 | 0.78 |
| sub6 | 0.74 | 0.77 | 0.78 | 0.83 | 0.73 |
| sub7 | 0.77 | 0.76 | 0.69 | 0.75 | 0.70 |
| sub8 | 0.87 | 0.89 | 0.87 | 0.91 | 0.71 |
| sub9 | 0.79 | 0.80 | 0.80 | 0.87 | 0.85 |
| 평균 | 0.75 | 0.76 | 0.74 | 0.78 | 0.75 |

|  |  |  |  |
| --- | --- | --- | --- |
| Sparse kernel  machine method[24] | Multi-ResNet[10] | Proposed method | Proposed method + Augmentation |
| 0.75 | 0.93 | 0.93 | 0.97 |
| 0.61 | 0.63 | 0.83 | 0.94 |
| 0.61 | 0.94 | 0.75 | 0.93 |
| 0.95 | 0.96 | 1.00 | 0.99 |
| 0.93 | 0.84 | 0.95 | 0.98 |
| 0.84 | 0.93 | 0.94 | 0.97 |
| 0.77 | 0.97 | 0.88 | 0.94 |
| 0.91 | 0.91 | 0.85 | 0.90 |
| 0.87 | 0.95 | 0.92 | 0.97 |
| 0.81 | 0.90 | 0.89 | 0.95 |

5. 결론 및 향후 연구

EEG 데이터는 교육, 생산, 마케팅, 엔터테인먼트 등 다양한 분야에서 사용될 수 있다. 이와 같이 BCI를 사용하기 위해서는 EEG 데이터의 정확한 분석이 필수적이다. 정확한 분석에는 EEG의 특성을 잘 반영할 수 있는 분석 모델과의 연동이 필요하다. 이미 Motor Imagery EEG를 분류하기 위해 CNN이나 LSTM, Resnet 등 여러 분류 알고리즘을 사용하여 연구가 진행되고 있다. 그러나, 기존 연구에서는 각 사람의 동작을 분류하는 정확도가 0.7~0.8 정도로 나타나거나, 0.9로 높은 경우에는 여러 사람에 대한 분류 정확도는 낮은 것으로 확인되었다. 이는 EEG 데이터가 사람마다 고유한 패턴을 가지고 있기 때문이다. 이를 해결하기 위해 본 연구에서는 제안하는 알고리즘 “SSD+FBCSP+1D-Augmentation 기법”에 FBCSP를 포함하여 정확도를 높였다.

원본 raw EEG 데이터는 이미 수집되어 있는 BCI competition IV 2b dataset를 사용하여 연구를 진행하였다. BCI competition IV 2b dataset에 전처리로 SSD, FBCSP를 적용한 후 1-Dimensional Augmentation을 사용하여 데이터 개수를 증가시키고 LSTM을 통한 classification을 실행하였다. 여러 가지 전처리와 인자를 변경하며 알고리즘을 실행해본 결과 위의 조합이 가장 높은 accuracy를 보였다. 전체 피실험자에 대한 데이터로 train/test를 진행했음에도 피실험자 별 accuracy에 비해 전체 accuracy가 소폭으로 감소한 것을 확인하였다. 향후 연구로 EEG-BCI 기구를 사용한 뇌파 데이터를 직접 수집하여 위와 동일한 전처리와 알고리즘을 적용하였을 때 결과를 본 후, 제안한 알고리즘에 대한 추가적인 변경과 수정을 진행할 예정이다.

참 고 문 헌

[1] M. A. Lopez-Gordo, D. Sanchez-Morillo, F. Pelayo Valle, “Dry EEG electrodes, Sensors” (Switzerland) 14 (2014)

[2] Sarah N.Abdulkader, Ayman Atia, Mostafa-Sami M.Mostafa, “Brain computer interfacing : Applications and challenges” (2014)

[3] Yang, Tao & Phua, Kok Soon & Yu, Juanhong & Selvaratnam, Thevapriya & Toh, Valerie & Ng, Wai & Ang, Kai & So, Rosa. (2019). Image-based Motor Imagery EEG Classification using Convolutional Neural Network. 1-4. 10.1109/BHI.2019.8834598.

[4] P. Wang, A. Jiang, X. Liu, J. Shang and L. Zhang, "LSTM-Based EEG Classification in Motor Imagery Tasks," in IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 26, no. 11, pp. 2086-2095, Nov. 2018, doi: 10.1109/TNSRE.2018.2876129.

[5] Mahsa Zeynali, Hadi Seyedarabi,

EEG-based single-channel authentication systems with optimum electrode placement for different mental activities, Biomedical Journal, Volume 42, Issue 4, 2019

[6] Tsipouras, M.G. Spectral information of EEG signals with respect to epilepsy classification. EURASIP J. Adv. Signal Process. 2019, 10 (2019). https://doi.org/10.1186/s13634-019-0606-8

[7] Teplan, Michal. (2002). Fundamental of EEG Measurement. MEASUREMENT SCIENCE REVIEW. 2.

[8] Wei Bin Ng et al 2019 IOP Conf. Ser.: Mater. Sci. Eng. 557 012032

[9] Ha KW, Jeong JW. Motor Imagery EEG Classification Using Capsule Networks. Sensors (Basel). 2019;19(13):2854. Published 2019 Jun 27. doi:10.3390/s19132854

[10] Shovon, Tanvir Hasan & Al Nazi, Zabir & Dash, Shovon & Hossain,. (2019). Classification of Motor Imagery EEG Signals with multi-input Convolutional Neural Network by augmenting STFT. 10.1109/ICAEE48663.2019.8975578.

[11] 박상훈, 김하영, 이다빛, 이상국."동작 상상 EEG 분류를 위한 필터 뱅크 기반 정규화 공통 공간 패턴". 정보과학회논문지, 44(6), 587-594.(2017)

[12] Halme, Hanna-Leena, and Lauri Parkkonen.“Comparing Features for Classification of MEG Responses to Motor Imagery.” PloS one vol. 11,12 e0168766. 16 Dec. 2016, doi:10.1371/journal.pone.0168766

[13] Leeb, Robert and Claudia Brunner. “BCI Competition 2008  Graz data set B.” (2008).

[14] Nikulin VV, Nolte G, Curio G. A novel method for reliable and fast extraction of neuronal EEG/MEG oscillations on the basis of spatio-spectral decomposition. Neuroimage. 2011;55(4):1528‐1535. doi:10.1016/j.neuroimage.2011.01.057

[16] Virgilio González, Carlos. (2018). Classification of Motor Imagery EEG Signals with CSP Filtering Through Neural Networks Models. 10.1007/978-3-030-04491-6\_10.

ㄴㄴㄴㄴㄴ

[17] Geun-Ho Park, Yu-Ri Lee, and Hyoung-Nam Kim, “ Filter Selection Method Using CSP and LDA for Filter-bank based BCI Systems”, Journal of The Institute of Electronics and Information Engineers Vol. 51, NO. 5, pp.1100, May 2014

[18] Kai Keng Ang, Zheng Yang Chin, Haihong Zhang and Cuntai Guan, "Filter Bank Common Spatial Pattern (FBCSP) in Brain-Computer Interface," 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), Hong Kong, 2008, pp. 2390-2397, doi: 10.1109/IJCNN.2008.4634130.

[19] Um, Terry T. et al. “Data Augmentation of Wearable Sensor Data for Parkinson’s Disease Monitoring Using Convolutional Neural Networks.” Proceedings of the 19th ACM International Conference on Multimodal Interaction - ICMI 2017 (2017): n. pag. Crossref. Web.

[21] Alhagry, Salma & Aly, Aly & El-Khoribi, Reda. “Emotion Recognition based on EEG using LSTM Recurrent Neural Network.” International Journal of Advanced Computer Science and Applications.8.10.14569/IJACSA.2017.081046.(2017)

[22] Wang, Boyu & Wong, Chi & Wan, Feng & Mak, Pengun & Mak, Pui-In & Vai, Mang. (2009). Comparison of different classification methods for EEG-based brain computer interfaces: A case study. 2009 IEEE International Conference on Information and Automation, ICIA 2009. 1416 - 1421. 10.1109/ICINFA.2009.5205138.

[23] Lawhern VJ, Solon AJ, Waytowich NR, Gordon SM, Hung CP, Lance BJ. EEGNet: a compact convolutional neural network for EEG-based brain-computer interfaces. J Neural Eng. 2018;15(5):056013. doi:10.1088/1741-2552/aace8c

[24] V. P. Oikonomou, S. Nikolopoulos, P. Petrantonakis and I. Kompatsiaris, "Sparse Kernel Machines for motor imagery EEG classification," 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Honolulu, HI, 2018, pp. 207-210, doi: 10.1109/EMBC.2018.8512195.

[25] Tabar YR, Halici U. A novel deep learning approach for classification of EEG motor imagery signals. J Neural Eng. 2017;14(1):016003. doi:10.1088/1741-2560/14/1/016003