

다중 클래스 SVM을 이용한 효과적인 손 형태 인식

변재희^{0,1}, 남윤영², 최유주^{3,*}

¹덕성여자대학교, ²유비쿼터스시스템연구센터, ³서울벤처정보대학원대학교

bjah@duksung.ac.kr, youngman@ajou.ac.kr, yjchoi@suv.ac.kr

Effective Hand-Pose Recognition using Multi-Class SVM

Jae-Hee Byeon^{0,1}, Yunyoung Nam², Yoo-Joo Choi³

¹Duksung Womens University, ²Center of Excellence for Ubiquitous System, ³Seoul University of Venture and Information

요 약

본 논문은 다중 클래스 SVM을 이용하여 손 형태를 효과적으로 인식할 수 있는 방법을 제시한다. 컴퓨터의 상호작용 연구가 활발해짐에 따라 컴퓨터가 인간의 행동을 얼마나 정확히 인식할 수 있는가에 대한 연구는 끊임없이 이루어지고 있다. 본 연구에서는 실시간으로 입력되는 손 영상에 대하여 색상(Hue)과 채도(Saturation)를 이용한 컬러모델을 기반으로 조명의 영향을 줄이며 손의 영역을 추출하고, 특히, 팔영역을 포함한 손영역이 촬영된 영상에서 손목 이후 부분을 제외한 손 영역만을 추출하도록 하였다. 손 형태를 인식하기 위하여 손 영역으로부터 손의 특징을 18 개의 특징값으로 표현하였고, 이를 통해 학습된 다중 클래스 SVM을 이용하여 손 형태를 인식하였다.

1. 서 론

최근 컴퓨터와 인간의 상호작용을 목표로 사람의 음성이나 손동작, 행동분석 등의 많은 연구들이 진행되고 있다. 시각적인 장치를 기반으로 하여 사용자는 좀 더 자유롭게 행동의 의미를 표현할 수 있고, 컴퓨터는 사람의 동작을 좀 더 정확히 인식할 수 있도록 특징을 정의하는데 연구의 초점들이 맞추어 지고 있다. 사람의 동작을 컴퓨터가 정확히 인지할 수 있게 된다면 범죄가 발생했을 때 그것을 예방하거나 범죄자를 검거하는데 걸리는 시간을 단축할 수 있을 것이다. 또한 사람의 동작을 인지함에 있어 부가적인 입력 장치를 쓰지 않고도 여러 가지 생활에 필요한 장비를 보다 섬세하게 다룰 수도 있어 사용자에게 보다 편리한 생활환경을 제공할 수 있을 것이다. 하지만 현재의 기술로는 컴퓨터가 동작을 인지하는데 있어서 시각적인 장치를 이용해 영상을 받아들일 때 빛의 영향에 따른 색의 변화로 인해 사람의 신체부위를 정확히 잡기 어렵다는 문제점과 시각적인 장치가 어느 곳에 있는가에 따라 입력되는 영상의 중심점, 각도차 등을 잡기 힘들다는 문제점 등이 있다[1].

특히, 손을 이용한 인터페이스 개발을 위하여 손 형태 인식에 대한 연구가 활발히 진행되어 왔다. 손 형태

를 인식하기 위해서는 각각의 형태에 대한 특징점을 추출해야 한다. 컴퓨터에 입력되는 사람 손의 공분산에 대한 고유벡터의 특징을 이용하여 중심축을 구해 특징값으로 삼고, 배경과 손 영역을 분리하여 관심영역으로 인식되는 부분의 단축/장축비, 손 영역의 조밀도를 통해 특징점을 추출해내는 방법등이 제시되어 왔다[2]. 또한, 손 형태 인식에 대한 인식률을 높이고 계산복잡도를 줄일 수 있도록 손의 특징값을 효과적으로 정의하는 다양한 연구들이 지속적으로 시도되고 있다.

본 연구는 조명환경에 안정적으로 사람의 여러 가지 신체부위 중, 손을 이용한 여러 가지 동작들을 인식하기 위하여, 손의 특성을 효율적으로 표현할 수 있는 특징값들을 정의하고 다중 클래스 SVM(Support Vector Machine)을 이용하여 각 동작을 학습하여 분류하는 방법을 제시하고, 실험을 통하여 그 정확성을 입증하였다.

2. 손 영역 추출

2.1 색상과 채도 기반 배경차분

배경학습 단계에서는 각 픽셀에 대한 색상(H) 요소에 대한 평균과 표준편차 값을 구하여, 평균화된 H 영상을 생성하여 평균화된 배경영상을 구축한다. H 요소에 대한 배경 영상의 훈련과정은 재귀적인 방법을 이용하여 시간 T_i 동안 업데이트 되는 평균영상과 표준편차 영상을 계산하는 방법으로 구한다. 배경영상의 각 픽셀에 대한 시

본 연구는 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 정보통신부의 유비쿼터스컴퓨팅및네트워크원천기반기술개발사업의 지원에 의한 것임.

* 교신저자 : [E mail] yjchoi@suv.ac.kr

간 t대의 H 성분의 평균과 표준편차의 크기는 식(1)과 같이 계산한다.

Mean Update :

$$\begin{aligned} \mu_n(0) &= H(0) && : t=0 \\ \mu_n(t) &= (1-\alpha)\mu_n(t-1) + \alpha H(t) && : t \geq 1 \end{aligned}$$

Variance Update :

$$\begin{aligned} \sigma^2_n(1) &= (H(1) - \mu_n(0))^2 && : t=1 \\ \sigma^2_n(t) &= (1-\alpha)\sigma^2_n(t-1) + \alpha(H(t) - \mu_n(t))^2 && : t \geq 2 \end{aligned} \quad (1)$$

또한 배경을 학습할 때 배경영역의 채도(S)의 요소도 식 (1)과 같은 방법으로 모델링한다.

이러한 배경 모델링 과정이 끝난 후 동적 객체가 포함된 T_i 이후부터의 연속 입력영상 RGB에 대하여 색상과 채도 기반의 컬러모델로 변환을 수행한다. 변환된 입력 H 성분과 평균화된 H 배경영상의 표준편차 영상과 차이를 비교하여, 평균화된 H 배경영상의 표준편차 값보다 큰 값을 보이는 영역을 동적 객체의 후보영역으로 간주하고, 입력영상에 대한 각 픽셀에 대한 채도의 분포도를 계산하여 배경영상과의 채도의 분포도 차이 값이 일정 임계값보다 큰 경우 동적 객체의 후보로 간주한다. 그림 1과 같이 위 두 조건을 모두 만족하는 영역을 동적 객체 영역으로 추출한다[3].

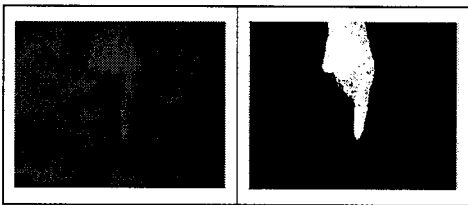


그림 1. 배경차분

2.2 손영역에 대한 방향기반 바운딩 박스(OBB) 추출

손목 영역을 제거한 손 영역을 제거하기 위해 우선적으로 동적객체의 검출영역을 전체 이미지 영역에 대해 실시하지 않고, 추출하고자 하는 특정 관심영역 방향기반 바운딩 박스(OBB: Object-Oriented Bounding Box)에 대하여 수행하였다. 우선, 동적객체의 OBB를 추출하기 위하여 동적 객체 영역의 구성 픽셀의 x, y 양방향 축에 대한 고유벡터와 고유값을 계산하고, 고유벡터에 의해 정의된 변환식을 정의하였다. 이를 모든 동적 객체 영역 픽셀에 적용함으로써, 동적 객체 영역의 주축을 x축으

로, 단축을 y축으로 재정렬하고, 정렬된 동적객체에 대한 바운딩 박스를 추출하였다. 이 과정을 단계별로 설명하면 다음과 같다. 입력영상에 대하여 추출하여 정규화하고자하는 동적객체의 x, y 방향에 대한 평균점 A 및 고유벡터의 장단축 방향을 구하여 평균점 A를 원점 O로 위치이동한 후 고유벡터의 장축방향인 B_u 방향을 x축으로 하고 단축방향인 B_v 방향을 y축으로 하여 회전처리를 하는데 이때 회전처리를 위해 특정벡터 u_1, u_n 을 이용한 변환행렬 W를 식 (2)과 같이 정의한다.

$$W = [u_1 \cdots u_n] \quad (2)$$

각 윤곽점의 좌표값을 x, y축으로 정렬시키는 것은 식 (3)와 같다.

$$y = W^T x \quad (3)$$

x, y축으로 주축을 이동한 후, 윤곽점에 대한 x 방향과 y방향에 대한 표준편차 S_x, S_y 를 기반으로 최대 가능한 OBB를 정의한다. OBB내에서 x축, y축의 ± 방향에 대해 원점으로부터 최대 거리를 가지는 윤곽점을 찾아 OBB를 조절한다. 이와 같이 표준편차를 이용한 최대 가능한 OBB를 미리 정의하고 OBB 내에서 윤곽점의 위치를 기반으로 최적화시키는 방법은 배경영상에 남아 있는 노이즈에 대한 ROI추출의 오류를 줄일 수 있다. [3]

2.3 손목 영역 제거

OBB의 x축을 min x 좌표부터 max x좌표로 옮겨 y축을 기준으로 손 영역의 두 경계점을 찾아 그 사이의 너비를 구한다. 다음 x 축 위치에서 경계 두 점 사이의 거리와 비교를 하여 10 % 이상으로 확장되었다면, 현재 접해있는 x 위치를 손목 위치로 잡고 손목점 이후의 x축에 있는 ROI를 손목을 제거한 손 영역으로 잡는다.[4]

3. 손 형태의 특징 정의

각각의 손 형태마다 그 특징을 나타낼 수 있는 특징값들이 있어야 하는데 이번 연구에서는 18개의 특징값들을 사용하여 각각의 손 형태를 분리하였다.

첫 번째와 두 번째 특징값은 고유값과 OBB에 대한 표준편차를 이용하여 각각 다음과 같이 구하였다.

- 첫 번째 특징값 : 고유값 대/OBB x에 대한 표준편차 값
- 두 번째 특징값 : 고유값 소/OBB y에 대한 표준편차 값

나머지 세 번째부터 열여덟 번째까지의 특징값은 그림 2과 같이 손 영역의 OBB를 16개 구역으로 나눈 후 각각의 영역마다 포함되는 픽셀의 개수를 구하여 만들었다.

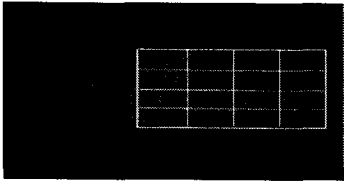


그림 2. 손 영역의 특징값 추출

4. SVM 기반 손형태 학습 및 인식

4.1 SVM 알고리즘

SVM은 이진분류를 위해 개발되었으며, 그림 3과 같이 분류 문제를 해결하기 위해 최적의 분리 경계면(hyperplane)을 제공한다. hyperplane을 중심으로 한쪽은 positive 클래스, 다른 한쪽은 negative 클래스로 나눈다. 처음의 SVM은 선형 분리 가능한 두 클래스로 구분하며 인접 데이터와의 거리가 최대가 되게 한다. 그림에서 경계와 가장 인접한 회색의 원과 사각형 데이터 점을 Support Vector(SV)라고 하며, 이 SV에는 두 클래스를 구분할 수 있는 모든 정보들이 포함되어 있기 때문에 이를 제외한 나머지 데이터는 필요 없게 되어 폐기된다. 최적의 hyperplane을 구하는 것(결정함수)은 식(4)과 같이 나타낼 수 있다.

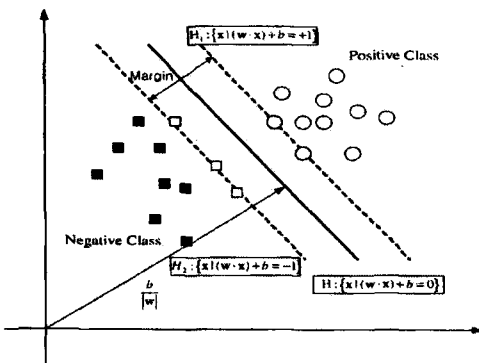


그림 3. SVM을 이용한 데이터의 이진분류

$$f(x) = \sum_{i=1}^n \alpha_i y_i(x_i, x) + b \quad (4)$$

선형 분리가 불가능한 데이터인 경우에는 kernel 함수를 이용하여 N차원의 입력공간의 데이터를 고차원의 특징 공간(Q차원)으로 변환함으로써 선형적으로 구별할 수 있으며 식(5)는 kernel 함수와 결정함수를 나타낸 것이다.[5]

$$K(x, y) = \phi(x) \cdot \phi(y)$$

$$f(x) = \sum_{i=1}^n \alpha_i y_i K(x, x_i) + b \quad (5)$$

SVM은 이진 분류를 위해 개발되었기 때문에 실제 환경에서 여러 클래스를 가지는 문제들을 해결하기에는 많은 어려움이 있다. 때문에 이러한 문제점들을 해결하기 위해 많은 전략들이 제시되었는데 그 중에 대표적인 것이 One-against-all 기법과 One-against-one 기법이다.

One-against-all 기법은 입력된 클래스의 개수만큼 SVM을 학습하는 방법이다. k개의 클래스가 입력되었을 때 k-1개의 SVM이 필요하며, 클래스 수가 늘어날수록 성능이 저하되고 각각의 클래스에 속하지 않는 입력이 주어지면 의미 없는 출력을 하게 된다.

One-against-one 기법은 k개의 클래스가 입력되었을 때 k(k-1)/2개의 SVM으로 구성되며 각각의 학습데이터는 두 개의 소속을 나타내는 데이터로만 구성된다. 각 학습에 사용되는 학습 데이터의 수가 적기 때문에 학습이 빠르다[6,7].

4.2 손 형태 특징 정의 파일

다중 클래스 SVM을 학습시키기 위해 만들어진 학습 데이터의 구성은 다음과 같다. 입력된 손 형태에 대하여 각각의 클래스 번호를 붙여준 후, 각각의 클래스 마다 18개의 특징값들을 구하여 학습 데이터를 구성한다. 예를 들어, 검지 손가락 하나를 펴고 있는 손 형태를 클래스 1이라고 한다면 이 손 형태에 특징값 18개를 추출하여 그림 4와 같이 구성한다.

1	:49.	03448276	2:36.	62962963	3:121	4:30	5:23	6:54	7:187	8:33	9:43	10:179	11:78	
1	:47.	75862069	2:37.	46153846	3:102	4:29	5:27	6:76	7:163	8:35	9:50	10:209	11:36	
1	:48.	39285714	2:37.	03703704	3:124	4:30	5:22	6:59	7:198	8:39	9:58	10:178	11:71	
1	:48.	48275862	2:37.	42307892	3:124	4:32	5:24	6:52	7:183	8:30	9:49	10:187	11:74	
1	:49.	75862069	2:36.	74074074	3:118	4:31	5:29	6:59	7:205	8:26	9:38	10:147	11:73	
1	:47.	92857143	2:36.	46428571	3:114	4:29	5:22	6:43	7:205	8:32	9:39	10:172	11:80	
1	:47.	75	2:37.	65384615	3:96	4:34	5:25	6:51	7:2207	8:34	9:34	10:172	11:68	12:107
1	:47.	80285714	2:37.	88461538	3:106	4:30	5:30	6:52	7:202	8:34	9:58	10:177	11:75	
2	:70.	16326531	2:35.	56523739	3:116	4:31	5:37	6:92	7:142	8:23	9:13	10:103	11:63	
2	:70.	10204082	2:35.	04347826	3:122	4:28	5:36	6:85	7:155	8:34	9:19	10:98	11:58	12:1
2	:69.	7755102	2:34.	56521739	3:116	4:32	5:27	6:98	7:149	8:19	9:15	10:107	11:47	12:1
2	:71.	30612245	2:34.	47826087	3:109	4:28	5:38	6:104	7:137	8:19	9:11	10:97	11:38	
2	:69.	30612245	2:35.	04347826	3:122	4:35	5:22	6:92	7:128	8:29	9:34	10:103	11:53	
2	:67.	73916667	2:35.	088091632	3:116	4:32	5:24	6:86	7:148	8:20	9:22	10:114	11:53	
2	:68.	47916667	2:36.	30434783	3:115	4:30	5:26	6:98	7:147	8:20	9:17	10:107	11:34	
2	:68.	83333333	2:34.	91304340	3:120	4:32	5:26	6:80	7:130	8:14	9:7	10:110	11:71	12:1
2	:69.	69387755	2:35.	43478261	3:103	4:32	5:41	6:79	7:128	8:21	9:3	10:98	11:27	12:1
2	:67.	64503333	2:35.	69563217	3:101	4:45	5:33	6:83	7:143	8:34	9:17	10:109	11:48	
2	:66.	07894737	2:32.	54545455	3:103	4:34	5:17	6:48	7:128	8:22	9:23	10:109	11:81	

클래스명 특징값번호 : 특징값

그림 4. 학습 데이터의 구성

손 형태 인식 알고리즘은 그림 5와 같이 요약 된다.

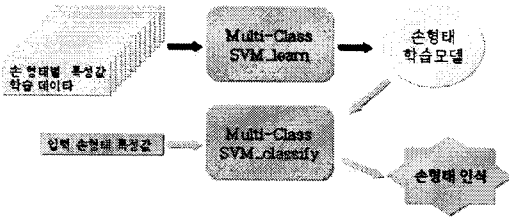


그림 5. 손 형태 인식 알고리즘

5. 실험 및 결과

입력된 영상의 동적객체를 추출하여 OBB를 추출하여 회전처리를 한 것은 그림 6과 같다. 그림 6의 (a)는 고유벡터와 고유값을 추출한 방향회전 전의 OBB이고 (b)는 (a)의 영상을 고유벡터의 방향을 x축과 y축에 맞게 회전 시킨 영상이다.

손 영역 이외의 부분이 입력되었을 때 손목 이후를 제거하여 손 영역만을 추출한 결과는 그림 7과 같다. 그림 7의 a는 손목을 포함한 영상이 입력되었을 때 동적객체만을 추출한 OBB이고 b는 입력된 영상에서 손목 영역을 제거하여 손 영역만을 추출한 영상이다.

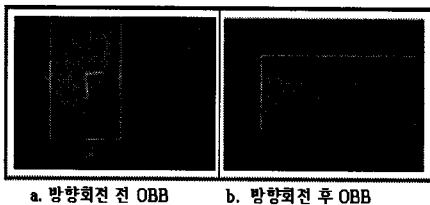


그림 6. OBB 이미지 추출과 방향회전 전/후 비교

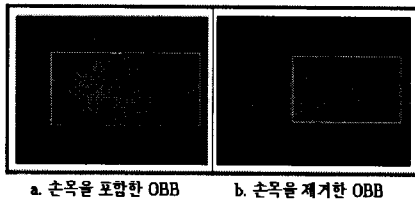


그림 7. 손목을 포함한 OBB와 손목을 제거한 OBB 비교

실험은 실험자 6인의 손 형태 9개를 입력받은 후, 한 클래스당 30개의 학습 데이터를 입력받아 총 1620개의 데이터를 다중 클래스 SVM으로 학습시켰다. 학습시킨 다중 클래스 SVM은 실험자 3인의 손 형태를 입력받아 한 클래스당 30개의 데이터를 만들어 총 810개의 데이터를 가지고 실험하였다. 실험결과는 표 1과 같이 96.29%의 인식률을 보였다.

표 1. 다중 클래스 SVM을 이용한 손 형태 인식률

Subgroup	Success rate(%)
1	100
2	100
3	100
4	100
5	66.6
6	100
7	100
8	100
9	100
Average	96.29

6. 결론

본 논문에서는 손 형태를 인식하기 위하여 18개의 주요 특징값을 정의하고 18개의 특징값에 따라 각기 서로 다른 9가지의 손 형태정보를 정의하였다. 이를 SVM을 이용하여 학습하여 학습 모델을 만들고 학습 모델을 이용한 손 인식 실험 결과 높은 손 형태 인식률을 얻었다.

향후 연구로는 16개의 손의 영역 정보를 추출시 미세한 손의 위치 이동에 강건한 인식률을 보일 수 있는 누적 계산 기법에 의한 특징 추출 기법을 적용하여 실험을 수행하고자 한다.

참고문헌

- [1] 이태경, 손동작 인식을 통한 인간-컴퓨터간의 상호작용에 관한 연구, 부산대학교 일반대학원 공학석사 학위논문, 2001
- [2] Hyoyoung Jang, Dae-Jin Kim, Jung-Bae Kim, Jin-Woo Jung, Kwang-Hyun Park and Z. Zenn Bien, Decomposition approach for hand-pose recognition, International Journal of HWRS, Vol.5, No.1, pp.21-25, 2004
- [3] 이제성, 조명영향을 최소화하는 동적객체추출에 관한 연구, 서울벤처정보대학원대학교 석사학위논문, 2006
- [4] A. Licsar, T. Sziranyi, User-adaptive hand gesture recognition system with interactive training, Image and Vision Computing, Vol. 23, No.12, pp. 1102-1114, 2005.
- [5] 황원우, 양보석, Multi-class SVM을 이용한 회전기계의 결함진단, 한국소음진동공학회논문집, 제14권, 제12호, pp.1233-1240, 2004
- [6] Hyunchul Ahn, Kyoung-jae Kim, Multiclass SVM, Model with Order Information, International Journal of Fuzzy Logic and Intelligent Systems, Vol. 6, No. 4, pp.331-334, December 2006
- [7] 고재필, Support Vector Machines을 이용한 다중 클래스 문제 해결, 정보과학회논문지:소프트웨어 및 응용, 제32권, 제12호, pp.1260-1270, 2005.12