



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2016년12월29일

(11) 등록번호 10-1690981

(24) 등록일자 2016년12월23일

(51) 국제특허분류(Int. Cl.)

G06K 9/18 (2006.01) *G06K 9/00* (2006.01)

(52) CPC특허분류

G06K 9/18 (2013.01)*G06K 9/00449* (2013.01)

(21) 출원번호 10-2015-7000030

(22) 출원일자(국제) 2013년07월23일

심사청구일자 2015년01월02일

(85) 번역문제출일자 2015년01월02일

(65) 공개번호 10-2015-0017755

(43) 공개일자 2015년02월17일

(86) 국제출원번호 PCT/US2013/051576

(87) 국제공개번호 WO 2014/018482

국제공개일자 2014년01월30일

(30) 우선권주장

201210258883.7 2012년07월24일 중국(CN)

13/947,412 2013년07월22일 미국(US)

(56) 선행기술조사문헌

JP08077294 A*

JP11232382 A*

JP2007052808 A*

JP07141471 A*

*는 심사관에 의하여 인용된 문헌

전체 청구항 수 : 총 21 항

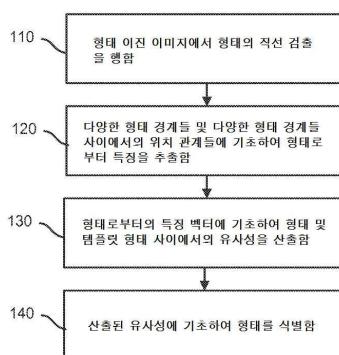
심사관 : 이종경

(54) 발명의 명칭 형태 인식 방법 및 디바이스

(57) 요 약

본 출원의 실시예들은 형태 인식 방법, 형태 인식 시스템, 및 형태들을 인식하기 위한 컴퓨터 프로그램 제품에 관한 것이다. 형태 인식 방법이 제공된다. 상기 방법은 상기 형태의 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 복수의 위치 관계들을 획득하기 위해 형태 이진 이미지에서의 형태의 직선 검출을 행하는 단계,

(뒷면에 계속)

대 표 도 - 도1

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 위치 관계들을 사용하여 상기 형태로부터 복수의 특징들을 추출하는 단계, 상기 복수의 특징들에 적어도 부분적으로 기초하여 상기 형태와 연관된 특징 벡터를 학습하는 단계, 및 상기 형태의 특징 벡터에 적어도 부분적으로 기초하여 상기 형태 및 복수의 템플릿 형태들의 각각의 것들 사이에서의 유사성들을 산출하는 단계, 및 상기 산출된 유사성들에 기초하여 상기 형태를 식별하는 단계를 포함한다.

명세서

청구범위

청구항 1

형태 인식 방법에 있어서,

형태의 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 복수의 위치 관계들을 획득하기 위해 형태 이진 이미지에서 상기 형태의 직선 검출을 행하는 단계로서;

시작점으로서 상기 형태상에서의 직선에 대한 설정점에 기초하여, 상기 직선상에서의 인접한 픽셀들에 대한 연결 구성요소 분석을 행하며 상기 직선의 연결 구성요소들을 획득하는 단계;

상기 직선의 상기 연결 구성요소를 상에서의 각각의 픽셀에 대한 상기 직선에 수직인 상기 연결 구성요소들의 픽셀들의 수를 획득하는 단계;

상기 직선의 상기 연결 구성요소들 상에서의 픽셀에 대한 상기 직선에 수직인 상기 연결 구성요소들의 픽셀들의 수에 기초하여 상기 직선의 평균 선 폭을 획득하는 단계;

상기 픽셀이 상기 직선의 상기 연결 구성요소들 상에 위치되는 선 폭이 상기 평균 선 폭에 적어도 부분적으로 기초하여 결정되는 임계 픽셀 선 폭보다 큰지를 결정하는 단계;

상기 픽셀이 상기 직선의 상기 연결 구성요소들 상에 위치되는 상기 선 폭이 상기 임계 픽셀 선 폭보다 큰 경우에, 상기 임계 픽셀 선 폭보다 큰 상기 픽셀의 선 폭이 위치되는 상기 선 폭에 대해 선 폭 제거 처리를 수행하는 단계를 포함하는, 상기 형태 이진 이미지에서 상기 형태의 직선 검출을 행하는 단계;

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들을 사용하여 상기 형태로부터 복수의 특징들을 추출하는 단계;

상기 복수의 특징들에 적어도 부분적으로 기초하여 상기 형태와 연관되고 상기 형태의 구조적 특징들을 나타내는 특징 벡터를 확립하는 단계;

상기 형태의 상기 특징 벡터에 적어도 부분적으로 기초하여 상기 형태 및 복수의 템플릿 형태들의 각각의 것들 사이에서의 유사성들을 산출하는 단계; 및

상기 산출된 유사성들에 기초하여 상기 형태를 식별하는 단계를 포함하는, 형태 인식 방법.

청구항 2

제 1 항에 있어서,

상기 형태 이진 이미지에서 상기 형태의 직선 검출을 행하는 단계는:

상기 선 폭 제거 처리를 겪은 상기 직선에 기초하여 상기 형태의 상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들을 획득하는 단계를 더 포함하는, 형태 인식 방법.

청구항 3

제 1 항에 있어서,

상기 임계 픽셀 선 폭보다 큰 상기 픽셀의 선 폭이 위치되는 상기 선 폭에 대해 상기 선 폭 제거 처리를 수행하는 단계는:

상기 직선의 상기 연결 구성요소들로부터 상기 임계 픽셀 선 폭보다 큰 상기 픽셀을 제거하는 단계;

제거된 상기 픽셀이 상기 직선 내에 있는지를 결정하는 단계; 및

제거된 상기 픽셀이 상기 직선 내에 있는 경우에, 제거된 상기 픽셀들의 양쪽 측면들 상에서 선분들을 병합하는 단계를 포함하는, 형태 인식 방법.

청구항 4

제 1 항에 있어서,

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들을 사용하여 상기 형태로부터의 상기 복수의 특징들을 추출하는 상기 단계 전에:

동일한 방향으로 연장되어 인접한 두 개의 형태 경계들이 설정 규칙에 기초하여 선분 병합을 요구하는지를 결정하는 단계; 및

상기 동일한 방향으로 연장되어 인접한 상기 두 개의 형태 경계들이 상기 설정 규칙에 기초하여 선분 병합을 요구하는 경우에, 상기 인접한 두 개의 형태 경계들에 대해 선분 병합을 수행하는 단계를 더 포함하는, 형태 인식 방법.

청구항 5

제 4 항에 있어서,

상기 동일한 방향으로 연장되어 인접한 상기 두 개의 형태 경계들이 상기 설정 규칙에 기초하여 선분 병합을 요구하는지를 결정하는 상기 단계는:

A) 상기 동일한 방향으로 연장하는 상기 두 개의 인접한 형태 경계들 사이에서의 거리를 결정하는 단계;

상기 거리를 설정된 제 1 임계 값에 비교하는 단계; 및

상기 비교 단계에 기초하여 선분 병합이 수행되어야 하는지를 결정하는 단계;

B) 상기 동일한 방향으로 연장하는 상기 두 개의 인접한 형태 경계들의 유사성을 결정하는 단계; 및

상기 유사성에 기초하여 선분 병합이 수행되어야 하는지를 결정하는 단계; 또는

C) A) 및 B)의 조합을 포함하는, 형태 인식 방법.

청구항 6

제 5 항에 있어서,

상기 동일한 방향으로 연장하는 상기 두 개의 인접한 형태 경계들의 상기 유사성을 결정하는 상기 단계는:

A) 상기 동일한 방향으로 연장하는 상기 두 개의 인접한 형태 경계들 사이에서의 각도를 결정하는 단계;

상기 각도를 설정된 제 2 임계 값에 비교하는 단계; 및

상기 비교하는 단계에 기초하여 선분 병합이 수행되어야 하는지를 결정하는 단계;

B) 상기 동일한 방향으로 연장하는 상기 두 개의 인접한 형태 경계들 사이에서의 평행 거리를 결정하는 단계;

상기 평행 거리를 설정된 제 3 임계 값에 비교하는 단계; 및

상기 비교하는 단계에 기초하여 상기 선분 병합이 수행되어야 하는지를 결정하는 단계; 또는

C) A) 및 B)의 조합을 포함하는, 형태 인식 방법.

청구항 7

제 2 항에 있어서,

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들을 사용하여 상기 형태로부터의 상기 복수의 특징들을 추출하는 상기 단계 전에, 상기 형태의 상기 복수의 형태 경계들의 문자 선분 필터링을 수행하는 단계를 더 포함하는, 형태 인식 방법.

청구항 8

제 7 항에 있어서,

상기 형태의 상기 복수의 형태 경계들의 상기 문자 선분 필터링의 상기 수행 단계는:

상기 형태의 상기 복수의 형태 경계들의 선분 길이들이 설정된 제 4 임계 값보다 작은지 또는 동일한지를 결정하는 단계; 및

상기 형태들의 상기 복수의 형태 경계들의 상기 선분 길이들이 상기 설정된 제 4 임계 값보다 작거나 또는 동일한 경우에, 그것의 선분 길이가 상기 제 4 임계 값보다 작거나 또는 동일한 형태 경계를 제거하는 단계를 포함하는, 형태 인식 방법.

청구항 9

제 8 항에 있어서,

상기 제 4 임계 값은 적어도:

상기 형태의 상기 복수의 형태 경계들에서 최대 연결 구성요소의 길이를 획득하며 상기 길이를 복수의 길이 간격들로 나누는 단계;

상기 각각의 연결 구성요소의 길이에 기초하여 상기 복수의 형태 경계들에서의 각각의 연결 구성요소를 상기 복수의 길이 간격들 중에서의 길이 간격으로 배분하는 단계;

상기 복수의 길이 간격들의 상기 길이들 및 상기 길이 간격들의 각각의 연결 구성요소들의 수에 기초하여 선분 길이 분포 히스토그램을 생성하는 단계; 및

상기 히스토그램에서의 y-좌표의 최고 값에 대응하는 상기 길이를 상기 제 4 임계 값으로서 할당하는 단계에 의해 설정되는, 형태 인식 방법.

청구항 10

제 1 항에 있어서,

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들을 사용하여 상기 형태로부터 상기 복수의 특징들을 추출하는 상기 단계는:

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들에 기초하여 상기 복수의 형태 경계들의 복수의 교차점들을 추출하는 단계;

상기 복수의 교차점들에 대한 정보를 획득하는 단계; 및

상기 복수의 형태 경계들로부터 수평으로 배향된 형태 경계들에 대한 정보 및 수직으로 배향된 형태 경계들에 대한 정보를 획득하는 단계를 포함하며,

상기 복수의 특징들에 적어도 부분적으로 기초하여 상기 형태와 연관된 상기 특징 벡터를 확립하는 상기 단계는 상기 복수의 교차점들에 대한 정보, 상기 수평으로 배향된 형태 경계들에 대한 정보, 및 상기 수직으로 배향된 형태 경계들에 대한 정보에 기초하여 상기 형태의 상기 특징 벡터를 획득하는 단계를 포함하는, 형태 인식 방법.

청구항 11

제 10 항에 있어서,

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들에 기초하여 상기 복수의 형태 경계들의 상기 복수의 교차점들을 추출하는 상기 단계 전에:

상기 형태의 영역을 결정하는 단계; 및

상기 형태의 상기 영역에 기초하여 상기 형태에 대한 정규화 처리를 수행하는 단계를 더 포함하는, 형태 인식 방법.

청구항 12

제 11 항에 있어서,

상기 형태에 대한 상기 정규화 처리를 수행하는 단계 후, 상기 형태를 복수의 블록들로 나누는 단계를 더 포함하며,

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들에 기초하여 상기 복수의 형태 경계들의 상기 복수의 교차점들을 추출하는 상기 단계 및 상기 복수의 교차점들에 대한 상기 정보를 획득하는 상기 단계, 및 상기 복수의 형태 경계들로부터 수평으로 배향된 형태 경계들에 대한 상기 정보 및 수직으로 배향된 형태 경계들에 대한 상기 정보의 획득 단계는:

형태의 블록들의 각각 내에서 상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들에 기초하여 상기 형태에서의 상기 형태의 블록으로부터 교차점을 추출하는 단계;

상기 형태의 블록에서 상기 형태들의 상기 교차점을 추출하는 단계;

상기 형태의 블록에서 상기 교차점에 대한 정보를 획득하는 단계; 및

상기 형태의 블록에서 수평으로 배향된 형태 경계들에 대한 정보 및 수직으로 배향된 형태 경계들에 대한 정보를 획득하는 단계를 포함하는, 형태 인식 방법.

청구항 13

제 1 항에 있어서,

상기 형태의 상기 특징 벡터에 적어도 부분적으로 기초하여 상기 형태 및 상기 복수의 템플릿 형태들의 상기 각각의 것들 사이에서의 상기 유사성을 산출하는 상기 단계는:

상기 형태의 상기 특징 벡터 및 상기 템플릿 형태의 특징 벡터에 대한 가장 가까운 거리 대 두 번째로 가까운 거리의 비를 산출하는 단계;

상기 비가 설정 비보다 크거나 또는 동일한 경우, 상기 형태 및 상기 가장 가까운 거리에 대응하는 상기 템플릿 형태 사이에서의 상기 유사성은 가장 높은 것으로 결정하는 단계; 및

상기 비가 상기 설정 비보다 작은 경우에, 상기 형태를 상기 템플릿 형태와 유사하지 않은 것으로 결정하는 단계를 포함하는, 형태 인식 방법.

청구항 14

형태 인식 시스템에 있어서,

적어도 하나의 프로세서로서:

형태의 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 복수의 위치 관계들을 획득하기 위해 형태 이진 이미지에서 상기 형태의 직선 검출을 행하는 것으로서;

시작점으로서 상기 형태상에서의 직선상에서의 설정점에 기초하여, 상기 직선 상에서의 인접한 픽셀들 상에서 연결 구성요소 분석을 행하며 상기 직선의 연결 구성요소들을 획득하는 것;

상기 직선의 상기 연결 구성요소들 상에서의 각각의 픽셀에 대해 상기 직선에 수직인 상기 연결 구성요소들의 픽셀들의 수를 획득하는 것;

상기 직선의 상기 연결 구성요소들 상에서의 픽셀에 대해 상기 직선에 수직인 상기 연결 구성요소들의 수에 기초하여 상기 직선의 평균 선 폭을 획득하는 것;

상기 픽셀이 상기 직선의 상기 연결 구성요소들 상에 위치되는 선 폭이 상기 평균 선 폭에 적어도 부분적으로 기초하여 결정되는 임계 픽셀 선 폭보다 큰지를 결정하는 것;

상기 픽셀이 상기 직선의 상기 연결 구성요소들 상에 위치되는 상기 선 폭이 상기 임계 픽셀 선 폭보다 큰 경우에, 상기 임계 픽셀 선 폭보다 큰 상기 픽셀의 상기 선 폭이 위치되는 상기 선 폭에 대해 선 폭 제거 처리를 수행하는 것을 포함하는, 상기 형태 이진 이미지에서 상기 형태의 직선 검출을 행하는 것,

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들을 사용하여 상기 형태로부터 복수의 특징들을 추출하는 것;

상기 복수의 특징들에 적어도 부분적으로 기초하여 상기 형태와 연관되고 상기 형태의 구조적 특징들을 나타내는 특징 벡터를 확립하는 것;

상기 형태의 상기 특징 벡터에 적어도 부분적으로 기초하여 상기 형태 및 복수의 템플릿 형태들의 각각

의 것들 사이에서의 유사성들을 산출하는 것;

상기 산출된 유사성들에 기초하여 상기 형태를 식별하는 것을 행하도록 구성된, 상기 적어도 하나의 프로세서; 및

상기 적어도 하나의 프로세서에 결합되며 상기 적어도 하나의 프로세서에 명령들을 제공하도록 구성된 메모리를 포함하는, 형태 인식 시스템.

청구항 15

제 14 항에 있어서,

상기 형태 이진 이미지에서 상기 형태의 상기 직선 검출을 행하는 것은:

상기 선 폭 제거 처리를 겪은 상기 직선에 기초하여 상기 형태의 상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들을 획득하는 것을 더 포함하는, 형태 인식 시스템.

청구항 16

제 15 항에 있어서,

상기 임계 픽셀 선 폭보다 큰 상기 픽셀의 상기 선 폭이 위치되는 상기 선 폭에 대해 상기 선 폭 제거 처리를 수행하는 것은:

상기 직선의 상기 연결 구성요소들로부터 상기 임계 픽셀 선 폭보다 큰 상기 픽셀을 제거하는 것;

제거된 상기 픽셀이 상기 직선 내에 있는지를 결정하는 것; 및

제거된 상기 픽셀이 상기 직선 내에 있는 경우에, 제거된 상기 픽셀들의 양쪽 측면들 상에서 선분들을 병합하는 것을 포함하는, 형태 인식 시스템.

청구항 17

제 14 항에 있어서,

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들을 사용하여 상기 형태로부터 상기 복수의 특징들을 추출하는 것은:

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들에 기초하여 상기 복수의 형태 경계들의 복수의 교차점들을 추출하는 것;

상기 복수의 교차점들에 대한 정보를 획득하는 것;

상기 복수의 형태 경계들로부터 수평으로 배향된 형태 경계들에 대한 정보 및 수직으로 배향된 형태 경계들에 대한 정보를 획득하는 것을 포함하며,

상기 복수의 특징들에 적어도 부분적으로 기초하여 상기 형태와 연관된 상기 특징 벡터를 확립하는 것은 상기 복수의 교차점들에 대한 정보, 상기 수평으로 배향된 형태 경계들에 대한 상기 정보, 및 상기 수직으로 배향된 형태 경계들에 대한 상기 정보에 기초하여 상기 형태의 상기 특징 벡터를 획득하는 것을 포함하는, 형태 인식 시스템.

청구항 18

제 17 항에 있어서,

상기 적어도 하나의 프로세서는 또한:

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들에 기초하여 상기 복수의 형태 경계들의 상기 복수의 교차점들을 추출하기 전에:

상기 형태의 영역을 결정하며;

상기 형태의 상기 영역에 기초하여 상기 형태에 대한 정규화 처리를 수행하도록 구성되는, 형태 인식 시스템.

청구항 19

제 18 항에 있어서,

상기 적어도 하나의 프로세서는 또한, 상기 형태에 대한 상기 정규화 처리를 수행한 후, 상기 형태를 복수의 블록들로 나누도록 구성되며;

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들에 기초하여 상기 복수의 형태 경계들의 상기 복수의 교차점들을 추출하는 것 및 상기 복수의 교차점들에 대한 상기 정보를 획득하는 것, 및 상기 복수의 형태 경계들로부터 수평으로 배향된 형태 경계들에 대한 상기 정보 및 수직으로 배향된 형태 경계들에 대한 상기 정보를 획득하는 것은:

형태의 블록들의 각각 내에서 상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들에 기초하여 상기 형태에서의 상기 형태의 블록으로부터 교차점을 추출하는 것;

상기 형태의 블록에서 상기 형태 경계들의 상기 교차점을 추출하는 것;

상기 형태의 블록에서 상기 교차점에 대한 정보를 획득하는 것; 및

상기 형태의 블록에서 수평으로 배향된 형태 경계들에 대한 정보 및 수직으로 배향된 형태 경계들에 대한 정보를 획득하는 것을 포함하는, 형태 인식 시스템.

청구항 20

제 14 항에 있어서,

상기 형태의 상기 특징 벡터에 적어도 부분적으로 기초하여 상기 형태 및 상기 복수의 템플릿 형태의 상기 각각의 것들 사이에서의 상기 유사성을 산출하는 것은:

상기 형태의 상기 특징 벡터 및 상기 템플릿 형태의 특징 벡터에 대한 가장 가까운 거리 대 두 번째로 가까운 거리의 비를 산출하는 것;

상기 비가 설정 비보다 크거나 또는 동일한 경우에, 상기 형태 및 상기 가장 가까운 거리에 대응하는 상기 템플릿 형태 사이에서의 상기 유사성을 가장 높은 것으로 결정하는 것; 및

상기 비가 상기 설정 비보다 작은 경우에, 상기 형태를 상기 템플릿 형태와 유사하지 않은 것으로 결정하는 것을 포함하는, 형태 인식 시스템.

청구항 21

형태들을 인식하기 위한 컴퓨터 프로그램이 기록된 컴퓨터 판독가능 기록 매체로서, 상기 컴퓨터 프로그램이 비-일시적 컴퓨터 판독 가능한 저장 매체에 구현되며 컴퓨터 명령들을 포함하는, 상기 컴퓨터 판독가능 기록 매체에 있어서,

상기 명령들은:

형태의 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 복수의 위치 관계들을 획득하기 위해 형태 이진 이미지에서 상기 형태의 직선 검출을 행하고;

시작점으로서 상기 형태상에서의 직선에 대한 설정점에 기초하여, 상기 직선상에서의 인접한 픽셀들에 대한 연결 구성요소 분석을 행하며 상기 직선의 연결 구성요소들을 획득하고;

상기 직선의 상기 연결 구성요소들 상에서의 각각의 픽셀에 대한 상기 직선에 수직인 상기 연결 구성요소들의 픽셀들의 수를 획득하고;

상기 직선의 상기 연결 구성요소들 상에서의 픽셀에 대한 상기 직선에 수직인 상기 연결 구성요소들의 픽셀들의 수에 기초하여 상기 직선의 평균 선 폭을 획득하고;

상기 픽셀이 상기 직선의 상기 연결 구성요소들 상에 위치되는 선 폭이 상기 평균 선 폭에 적어도 부분적으로 기초하여 결정되는 임계 픽셀 선 폭보다 큰지를 결정하고;

상기 픽셀이 상기 직선의 상기 연결 구성요소들 상에 위치되는 선 폭이 상기 임계 픽셀 선 폭보다 큰 경우에, 상기 임계 픽셀 선 폭보다 큰 상기 픽셀의 선 폭이 위치되는 상기 선 폭에 대해 선 폭 제거 처리를 수행하

고;

상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 상기 위치 관계들을 사용하여 상기 형태로부터 복수의 특징들을 추출하고;

상기 복수의 특징들에 적어도 부분적으로 기초하여 상기 형태와 연관되고 상기 형태의 구조적 특징들을 나타내는 특징 벡터를 확립하고;

상기 형태의 상기 특징 벡터에 적어도 부분적으로 기초하여 상기 형태 및 복수의 템플릿 형태들의 각각의 것들 사이에서의 유사성들을 산출하며;

상기 산출된 유사성들에 기초하여 상기 형태를 식별하기 위한 것인, 컴퓨터 판독가능 기록 매체.

발명의 설명

기술 분야

[0001]

다른 출원들에 대한 상호 참조

[0002]

본 출원은, 여기에 모든 목적들을 위해 참조로서 통합되는, 2012년 7월 24일에 출원된, 형태 인식 방법 및 디바이스라는 제목의, 중화 인민 공화국 특허 출원 번호 제201210258883.7호에 대한 우선권을 주장한다.

[0003]

본 출원은 형태 인식 방법 및 디바이스에 관한 것이다.

배경 기술

[0004]

"패턴 인식"은 다양한 형태들의 대표적인 것들 및 현상들을 설명, 식별, 분류, 및 해석하기 위해 이러한 것들 및 현상들에 대한 정보(수치, 텍스트, 및 논리 관계)를 프로세싱 및 분석하는 것에 관한 것이다. 컴퓨터 기술이 개발됨에 따라, 컴퓨터들은 이벤트들 또는 프로세스들을 식별 및 분류하기 위해 패턴 인식에 적용되기 시작하였다. 식별된 이벤트들 또는 프로세스들은 문자들, 사운드들, 또는 화상들과 같은 구체적인 오브젝트들을 포함할 수 있거나, 또는 식별된 이벤트들 또는 프로세스들은 상태들 또는 정도들과 같은 추상적 오브젝트들을 포함할 수 있다.

[0005]

예를 들면, 컴퓨터들은 문자 인식을 수행하기 위해 사용되고 있다. 일 예로서 광학적 문자 인식을 사용할 때, 광학적 문자 인식(OCR)은 그에 의해 전자 장비(예로서, 스캐너 또는 디지털 카메라)가 종이 상에 인쇄된 문자들을 검사하고 어둡기 및 밝기의 패턴들을 검출함으로써 문자의 형태들을 결정하며 그 후 결정된 형태들을 컴퓨터 문자들로 변환하기 위해 문자 인식 방법을 사용하는 프로세스를 나타낸다. 다시 말해서, 텍스트 재료들은 이미지 파일들을 생성하기 위해 스캐닝되며 그 후 이미지 파일들은 문자 및 배치 정보를 획득하기 위해 분석된다. 대다수의 텍스트 문자들은 OCR을 사용하여 인식될 수 있다.

발명의 내용

해결하려는 과제

[0006]

그러나, 문자 인식을 위한 컴퓨터들의 현재 애플리케이션은 여전히 몇몇 한계들을 가진다. 예를 들면, 기존의 OCR 기술은 종종 텍스트 재료들로부터 스캐닝된 이미지 파일들에서 형태들을 매우 정확하게 인식하지 않는다. 컴퓨터가 형태를 접할 때, 기존의 OCR 기술은 종종 가비지 문자들을 생성하며 형태를 정확하게 식별할 수 없다.

과제의 해결 수단

[0007]

본 발명에 따른 형태 인식 방법은 상기 형태의 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 복수의 위치 관계들을 획득하기 위해 형태 이진 이미지에서의 형태의 직선 검출을 행하는 단계, 상기 복수의 형태 경계들 및 상기 복수의 형태 경계들 사이에서의 위치 관계들을 사용하여 상기 형태로부터 복수의 특징들을 추출하는 단계, 상기 복수의 특징들에 적어도 부분적으로 기초하여 상기 형태와 연관된 특징 벡터를 확립하는 단계, 및 상기 형태의 특징 벡터에 적어도 부분적으로 기초하여 상기 형태 및 복수의 템플릿 형태들의 각각의 것들 사이에서의 유사성들을 산출하는 단계, 및 상기 산출된 유사성들에 기초하여 상기 형태를 식별하는 단계를 포함한다.

발명의 효과

[0008]

본 발명에 따른 형태 인식 방법은 형태 경계 경계들의 정확하고, 선명한 추출을 가능하게 하며, 형태 인식의 정확도를 증가시킬 수 있다.

도면의 간단한 설명

[0009]

도 1은 형태 인식 프로세스의 실시예의 흐름도.

도 2는 형태 인식 프로세스의 또 다른 실시예의 흐름도.

도 3a는 형태의 실시예의 다이어그램.

도 3b는 수평 직선의 실시예의 다이어그램.

도 4는 형태 특징들 다이어그램의 실시예를 도시하는 도면.

도 5 및 도 6은 거리 유사성을 통해 식별되는 형태들의 실시예들의 다이어그램들.

도 7은 형태 직선 검출 프로세스의 실시예의 흐름도.

도 8은 형태 이미지 프로세스로부터 연결된 구성요소들의 추출의 실시예의 흐름도.

도 9a는 형태 직선 검출에서 선분 병합(line segment merging)에서의 거리 결정의 실시예의 다이어그램.

도 9b는 병합된 선분의 실시예의 다이어그램.

도 10은 형태 직선 검출에서 선분 병합에서의 각도 결정의 실시예의 다이어그램.

도 11은 형태 직선 검출에서 선분 병합에서의 평행선 거리 결정의 실시예의 다이어그램.

도 12는 형태 이미지 문자 선분 추출 프로세스의 실시예의 흐름도.

도 13은 형태 직선 검출에서 문자 선분 필터링의 실시예의 다이어그램.

도 14는 형태 카테고리 인식 프로세스의 실시예의 흐름도.

도 15는 형태 특징들 프로세스의 추출의 실시예의 흐름도.

도 16은 형태 카테고리 인식을 겪은 다중-레벨, 계층화된 형태의 다이어그램.

도 17은 형태상에 수행된 직선 검출에 따른 출력 결과들의 차트.

도 18은 형태 출력 결과상에 수행된 특징점 추출에 따른 특징점들의 다이어그램.

도 19는 다양한 모델들 프로세스에 대한 형태의 유사성들의 산출의 실시예의 흐름도.

도 20은 형태 모델 프로세스의 확립의 실시예의 흐름도.

도 21은 형태 인식 디바이스의 실시예의 구조적 블록 차트.

도 22는 형태 인식을 위한 시스템의 실시예의 구조도.

발명을 실시하기 위한 구체적인 내용

[0010]

본 발명의 다양한 실시예들이 다음의 상세한 설명 및 첨부한 도면들에 개시된다.

[0011]

본 발명은 프로세스; 장치; 시스템; 재료의 구성; 컴퓨터 관리 가능한 저장 매체 상에 구체화된 컴퓨터 프로그램 제품; 및/또는 프로세서에 결합된 메모리 상에 저장되고 및/또는 그것에 의해 제공된 명령들을 실행하도록 구성된 프로세서와 같은, 프로세서를 포함하여, 다수의 방식들로 구현될 수 있다. 본 명세서에서, 이를 구현들, 또는 본 발명이 취할 수 있는 임의의 다른 형태는 기술들로서 불리울 수 있다. 일반적으로, 개시된 프로세스들의 단계들의 순서는 본 발명의 범위 내에서 변경될 수 있다. 달리 서술되지 않는다면, 태스크를 수행하도록 구성되는 바와 같이 설명된 프로세서 또는 메모리와 같은 구성요소는 주어진 시간에 태스크를 수행하도록 임시로 구성되는 일반 구성요소 또는 태스크를 수행하기 위해 제조되는 특정 구성요소로서 구현될 수 있다. 여기에 사용된 바와 같이, 용어('프로세서')는 컴퓨터 프로그램 명령들과 같은, 데이터를 프로세싱하도록 구성된 하나 이

상의 디바이스들, 회로들, 및/또는 프로세싱 코어들을 나타낸다.

[0012] 본 발명의 하나 이상의 실시예들의 상세한 설명이 본 발명의 원리들을 예시하는 첨부한 도면들과 함께 이하에 제공된다. 본 발명은 이러한 실시예들과 관련되어 설명되지만, 본 발명은 임의의 실시예로 제한되지 않는다. 본 발명의 범위는 청구항들에 의해서만 제한되며 본 발명은 다수의 대안들, 수정들, 및 등가물들을 포함한다. 다수의 특정한 세부사항들이 본 발명의 철저한 이해를 제공하기 위해 다음의 설명에 제시된다. 이들 세부사항들은 예를 위해 제공되며 본 발명은 이들 특정 세부사항들의 일부 또는 모두 없이 청구항들에 따라 실시될 수 있다. 명료함을 위해, 본 발명에 관련된 기술 분야들에 알려져 있는 기술적 자료는 본 발명이 불필요하게 모호해지지 않도록 상세히 설명되지 않았다.

[0013] 도 1은 형태 인식 프로세스의 실시예의 흐름도이다. 프로세스(100)는 도 22의 서버(2220)에 의해 구현될 수 있으며 다음을 포함한다.

[0014] 110에서, 서버는 형태 이진 이미지에서의 형태의 직선 검출을 행하며 상기 형태의 복수의 형태 경계들 및 다양한 형태 경계들 사이에서의 복수의 위치 관계들을 획득한다.

[0015] 형태에서, 통상적으로, 선분(line segment)들은 선명하며 똑바르다. 선분들은, 직선 검출의 대상이 될 때, 형태의 복수의 형태 경계들 및 복수의 형태 경계들 사이에서의 위치 관계들을 산출한다. 상기 형태에서의 형태 경계들은 형태에서의 선분들을 나타낸다. 직선 검출은 허프 변환(Hough Transform) 또는 또 다른 잘 알려진 기술에 의해 구현될 수 있다.

[0016] 120에서, 서버는 복수의 형태 경계들 및 상기 형태 경계들 사이에서의 위치 관계들에 기초하여 상기 형태로부터 특징들의 세트를 추출하며, 상기 형태로부터 특징 벡터를 확립한다. 특징 벡터는 형태의 구조적 특징들을 나타낸다.

[0017] 130에서, 서버는 형태의 특징 벡터에 기초하여 상기 형태 및 템플릿 형태들 사이에서의 유사성들을 산출한다.

[0018] 몇몇 실시예들에서, "템플릿 형태들"은 서버에 저장된 형태들이다. 형태 모델들은 샘플 형태 이미지를 상에 직선 검출 및 특징 추출을 수행함으로써 확립된다. 몇몇 실시예들에서, "템플릿 형태들"은 수동으로 생성된다. 몇몇 실시예들에서, "템플릿 형태들"은 다른 방식들로 생성된다는 것이 이해된다.

[0019] 140에서, 서버는 산출된 유사성들에 기초하여 형태를 식별한다.

[0020] 몇몇 실시예들에서, 서버는 선분들 및 형태의 선분들 사이에서의 관계들을 획득하기 위해 형태의 구조적 특징들에 기초하여, BMP 파일들, TIF 파일들, JPEG 파일들, GIF 파일들, PNG 파일들 등과 같은 형태 이진 이미지에서의 형태상에서 직선 검출을 수행한다. 다시 말해서, 서버는 형태의 다양한 형태 경계들 및 상기 다양한 형태 경계들 사이에서의 위치 관계들을 획득한다. 그 후, 서버는 형태의 형태 경계들 및 위치 관계들에 관한 정보에 기초하여 상기 형태상에서의 특징을 추출하며 상기 형태의 구조적 특징들을 획득하기 위해 상기 형태의 특징 벡터를 획득한다. 그 뒤에, 형태는 논의 중인 형태와 비교하여, 어떤 템플릿이 형태에 대한 가장 큰 유사성을 갖는지, 다시 말해서 가장 유사한 구조를 결정하기 위해 형태의 구조적 특징들에 기초하여 템플릿들에 대하여 매칭된다. 형태의 카테고리는 매칭 결과들에 기초하여 결정된다. 종래의 시스템들은 보통 이미지 파일들 또는 형태 이미지들에서의 형태들을 바르게 식별할 수 없으며, 컴퓨터로의 편리한 및 빠른 입력을 위한 이미지 파일들에서의 형태들 또는 형태 이미지들에서의 형태들을 정확하게 식별할 수 없다. 편리함을 위해, "이미지 파일들 또는 형태 이미지들에서의 형태들"은 이하에서 총괄하여 "형태 이미지들"로서 불리운다.

[0021] 도 2는 형태 인식 프로세스의 또 다른 실시예의 흐름도이다. 프로세스(200)는 도 22의 서버(2220)에 의해 구현될 수 있으며 다음을 포함한다:

[0022] 첫 번째로, 프로세스(200)는 형태상에 직선 검출을 구현하기 위해 연결된 구성요소 분석을 이용하여 획득된 형태 경계들 상에서 추가 최적화 처리를 구현한다. 두 번째로, 형태 특징 추출을 수행하기 전에, 프로세스(200)는 형태 경계들 상에서 문자 선분 필터링을 수행한다. 더욱이, 프로세스(200)는 특징 교차점들을 추출함으로써 형태 특징 추출을 달성한다. 또한, 프로세스(200)는 형태 및 템플릿 형태들에 대한 가장 가까운 거리 대 두 번째로 가까운 거리의 비를 결정함으로써 정확한 형태 인식을 달성한다.

[0023] 프로세스(200)는 "田"의 형태에서의 간단한 2×2 형태의 예를 사용하여 설명된다.

[0024] 210에서, 서버는 직선 검출을 구현하기 위해, 형태의 다양한 형태 경계들을 획득하기 위해 및 다양한 형태 경계들 사이에서의 위치 관계를 획득하기 위해 형태 이진 이미지에서의 형태의 각각의 선분에 대한 연결 구성요소

분석을 수행한다.

[0025] 몇몇 실시예들에서, 서버는 각각의 선분에 대한 연결 구성요소 분석을 행하며, 다시 말해서, 상기 형태의 각각의 직선에 대해, 서버는 시작점으로서 각각의 직선상에서 설정점(set point)을 사용한다(예를 들면, 설정점은 시작점, 종점, 또는 직선상에서의 임의의 다른 설정점에 대응한다). 연결 구성요소 분석은 직선의 연결 구성요소들을 획득하기 위해, 시작점의 인접 픽셀들, 상기 시작점에 인접한 것으로 간주되는 픽셀(들)의 인접 픽셀들 등 상에서 수행된다. 그 후, 서버는 각각의 직선의 연결 구성요소들 상에서 각각의 픽셀에 대한 각각의 직선에 수직인 연결 구성요소들의 수를 획득한다. 서버는 직선의 연결 구성요소들 상에서 각각의 픽셀에 대한 각각의 직선에 수직인 연결 구성요소들의 수에 기초하여 각각의 직선의 평균 선 폭을 획득한다. 서버는 각각의 픽셀이 직선의 연결 구성요소들 상에 위치되는 선 폭이 평균 선 폭보다 큰지를 결정한다. 선 폭이 평균 선 폭보다 큰 경우에, 서버는 평균 선 폭보다 큰 픽셀들이 위치되는 선 폭 상에서 선 폭 제거 처리를 수행한다. 선 폭 제거 처리를 수행한 후, 서버는 선 폭 제거 처리를 겪은 각각의 직선에 기초하여 형태의 형태 경계들을 획득한다. 몇몇 실시예들에서, 실질적으로 동시에, 서버는 또한 연결 구성요소들 사이에서의 위치 관계들에 기초하여, 선분 시작점에서의 형태 경계(B)에 수직인 형태 경계(A)와 같이, 다양한 형태 경계들 사이에서의 위치 관계들을 획득한다.

[0026] 검출된 직선들은 주로 수평 형태 선들 및 수직 형태 선들이기 때문에, 위치 관계들은: 수평 선들 사이에서의 위/아래 관계들, 수직 선들 사이에서의 좌측/우측 관계들, 및 코너에서 상호 교차하거나 또는 교차하는, 수평 및 수직 선들 사이에서의 교차 관계들을 포함한다. 이를 위치 관계들은 직선들의 종점 좌표들에 기초하여 결정될 수 있다.

[0027] 몇몇 실시예들에서, 서버가 평균 선 폭보다 큰 픽셀들이 위치되는 선 폭 상에서 선 폭 제거 처리를 수행하는 경우에, 서버는 직선의 연결 구성요소들로부터 평균 선 폭보다 큰 픽셀들을 제거한다. 그 뒤에, 서버는 제거된 픽셀들이 직선 내에 위치되는지를 결정한다. 제거된 픽셀들이 직선 내에 위치되는 경우에, 서버는 제거된 픽셀들의 양쪽 측면들 모두에 대해 선분 병합을 수행한다.

[0028] 상기 프로세스(200)를 수행함으로써, 서버는 비교적 열악한 품질의 형태 이미지들, 특히 흐릿하며 거친 형태 이미지들을 효과적으로 프로세싱할 수 있다.

[0029] 도 3a는 형태의 실시예의 다이어그램이다. 형태는 "田" 형 형태이다. 이하의 예는 "田" 형 형태에서 제 2 수평 선이다. 연결 구성요소 분석에 의해 구현된 형태 직선 검출 프로세스가 설명된다. 도 3b는 수평 직선의 실시예의 다이어그램이다.

[0030] 제 2 수평 선의 고정 점(예를 들면, 도 3b에서의 픽셀 1)은 직선의 연결 구성요소 분석을 행하기 위해 시작점으로서 취해진다. 예를 들면, 서버는 제 2 수평 직선의 시작점 픽셀(다시 말해서, 픽셀 1)로서 현재 픽셀을 설정하며 수평 방향(좌측 또는 우측)으로 현재 픽셀에 인접한 다음 픽셀이 존재하는지를 결정한다. 이 예에서, 연결 구성요소 분석이 또한 다른 픽셀 1 상에서 수행된다. 다시 말해서, 연결 구성요소 분석은 상이한 선 폭들을 수용하기 위해 픽셀 1들 또는 각각의 픽셀 1 양쪽 모두 상에서 수행된다. 상기 예는 다음 픽셀이 시작점 픽셀의 우측면 상에 위치된 것을 도시한다. 수평 방향으로 형태 픽셀에 인접한 다음 픽셀이 존재하는 경우에, 서버는 연결 구성요소 영역에 다음 픽셀을 부가하며 부가된 다음 픽셀을 현재 픽셀로 간주한다. 서버는 다시 인접한 픽셀이 수평 방향으로 존재하는지를 결정하며, 현재 픽셀에 인접한 어떤 픽셀도 존재하지 않을 때까지 결정을 계속한다. 따라서, 서버는 직선 연결 구성요소(예를 들면, 도 3b의 픽셀들(1-2-3-4-5-6-7-8))를 획득한다. 몇몇 실시예들에서, 형태가 간단하며 명확할 때, 본 예에서처럼, 제 2 수평 선의 연결 구성요소는 전체 선을 커버한다. 몇몇 실시예들에서, 몇몇 복잡한 형태들에서, 직선상에서 수행된 연결 구성요소 분석은 복수의 연결 구성요소들을 생성한다.

[0031] 연결 구성요소 상에서의 각각의 픽셀에 대해, 서버는 직선상에서의 하나의 점으로서 선에 수직인 픽셀들을 고려한다. 예를 들면, 도 3b의 픽셀 1에서 수직 방향에서의 픽셀들 모두는 상이한 선 폭들을 수용하기 위해 픽셀 1로서 표시된다. 이러한 상황에서, 직선의 연결 구성요소 상에서 각각의 픽셀에 수직인 연결 구성요소들의 픽셀들의 수를 획득하는 것이 가능하다. 몇몇 실시예들에서, 이러한 동작은 직선상에서 연결 구성요소 분석을 수행하기 전에 직선상에서 수행된다. 이때, 직선상에서 각각의 픽셀에 대한 직선에 수직인 연결 구성요소의 픽셀들의 수를 획득하는 것은 수직 연결 구성요소들이 제 2 수평 선의 연결 구성요소에 대해 획득된 후 가능하다.

[0032] 선의 평균 폭의 산출은 선에 수직인 연결 구성요소들의 픽셀들의 수에 기초한다. 상기 예에서, 선의 평균 폭이 제 2 수평 선 상에서의 수직 픽셀들의 평균 수에 대응한다. 평균 선 폭은 N(픽셀들)이라고 가정하자. 연결 구성

요소 상에서의 각각의 픽셀 점에 대해, 연결 구성요소의 위치, 다시 말해서 현재 위치의 선 폭(수직 픽셀들의 수)이 임계 값보다 큰 경우에, 또는 선 폭 대 N의 비가 미리 정의된 임계 값을 초과하는 경우에, 선 폭 제거 처리가 픽셀 점이 위치되는 선 폭 상에서 수행된다. 예를 들면, 초과 선 폭을 가진 픽셀 점들이 제거되거나, 또는 픽셀 점들이 연결 구성요소들로부터 제거된다. 제거된 픽셀 점들이 직선 내에 위치되는 경우에, 직선은 두 개의 세그먼트들로 분할된다. 그 후, 두 개의 선분들이 병합된다. 일 예로서, 도 3b를 사용하여, N은 수평 직선의 평균 선 폭이며, 이것은 $(2+3+3+4+2+3+3+1)/8 = 21/8 = 2.625$ 이다. 선 폭 대 N의 비의 임계 값이 1.2라고 가정하면, $1.2 \times 2.625 = 3.15$ 이다. 다시 말해서, 수직 픽셀들의 수가 3.15를 초과하면, 픽셀 제거가 수행된다(예를 들면, 도 3b에서 픽셀들(4)). 제거된 픽셀이 직선의 중간(예를 들면, 도 3b의 픽셀 4)에 나타나는 경우에, 직선은 두 개의 직선들(1-2-3) 및 (5-6-7-8)로 분할된다. 따라서, 잡음, 문자들, 및 수직 선들의 효과들을 제거하는 것이 가능하다.

[0033] 몇몇 실시예들에서, 수직 선들의 직선 검출은 상기 설명된 수평 선들의 직선 검출에 대응한다.

[0034] 도 2로 가면, 220에서, 서버는 문자들의 선들로 간주되는 선들을 제거하기 위해 획득된 형태의 다양한 형태 경계들 상에서 문자 선분 필터링을 수행한다. 몇몇 실시예들에서, 문자 선분들은 문자 선분들이 형태 경계들과 쉽게 혼동될 수 있기 때문에 걸러내어진다.

[0035] 문자 선분 필터링은 형태의 다양한 형태 경계들의 선분 길이들이 설정된 제 4 임계 값보다 작은지 또는 동일한지를 결정하는 것을 포함한다. 형태의 다양한 형태 경계들의 선분 길이들이 설정된 제 4 임계 값보다 작거나 또는 동일한 경우에, 서버는 그것의 선분 길이가 제 4 임계 값보다 작거나 또는 동일한 형태 경계들을 제거한다.

[0036] 동작(220)에서, 제 4 임계 값이 많은 방식들로 설정된다. 예를 들면, 제 4 임계 값은 테스트 결과들, 경험적 값들에 기초하여 설정되거나, 또는 몇몇 다른 적절한 방식으로 설정된다. 몇몇 실시예들에서, 제 4 임계 값은 다음과 같이 설정된다: 형태의 다양한 형태 경계들에서 최대 연결 구성요소의 길이를 획득하며 최대 연결 구성요소의 길이를 복수의 길이 간격들로 나누며, 각각의 연결 구성요소의 길이에 기초하여 다양한 형태 경계들에서의 각각의 연결 구성요소를 복수의 길이 간격들 중에서의 길이 간격에 배분하고, 복수의 길이 간격들의 길이들 및 길이 간격들의 연결 구성요소들의 수에 기초하여 선분 길이 분포 히스토그램을 생성하며, 상기 히스토그램에서의 y-좌표의 최고점에 대응하는 길이를 제 4 임계 값으로 확립한다.

[0037] 몇몇 실시예들에서, 획득된 형태의 형태 경계들은 많은 문자 선분들을 가진다. 통상적으로, 선분들의 길이(문자 획들에 대응하는)는 직선들 또는 형태 경계들에 비교하여 작으며 사전 설정된 임계 값을 갖고 걸러내어질 수 있다. 몇몇 실시예들에서, 사전 설정된 값은 10 내지 20 픽셀들 사이에 있지만; 사전 설정된 임계 값은 다른 값들로 설정될 수 있다. 문자 선분 필터링의 사용은 형태 인식의 정확도를 효과적으로 개선하며 형태 인식에 대한 문자들의 효과들을 감소시킨다.

[0038] 상기 문자 선분 필터링 동작은 선택적 동작이며 몇몇 형태들이 순수 형태 이미지들(다시 말해서, 형태들 전용 또는 텍스트 또는 문자들 없는 형태들)에 있는 경우에 생략될 수 있다.

[0039] 230에서, 서버는 형태상에서 특징 추출을 수행하며 다양한 형태 경계들이 문자 세그먼트 필터링을 겪고 다양한 형태 경계들 사이에서의 위치 관계들이 획득된 후 다양한 형태 경계들에 기초하여 형태로부터 특정 벡터들을 획득한다.

[0040] 몇몇 실시예들에서, 서버는 형태 경계들의 다양한 교차점들을 추출하고, 상기 다양한 교차점들에 대한 정보를 획득하며, 다양한 형태 경계들 및 상기 다양한 형태 경계들 사이에서의 위치 관계들에 기초하여 다양한 형태 경계들 중에서 수평으로 배향된 형태 경계들에 대한 정보 및 수직으로 배향된 형태 경계들에 대한 정보를 획득한다. 서버는 상기 다양한 교차점들에 대한 정보, 상기 수평으로 배향된 형태 경계들에 대한 정보, 및 상기 수직으로 배향된 형태 경계들에 대한 정보에 기초하여 형태의 특징 벡터를 획득한다.

[0041] 예를 들면, 도 4에 도시된 바와 같이, 간단한 "田" 형 형태는 추출될 수 있는 총 9개 유형들의 교차점들(또한 특징점들로서 불리우는)을 가진 공통 형태이다. 형태 특성들에 기초하여, 9개 유형들의 추출 가능한 특징점들은: 북서쪽 특징점 유형(도 4에서의 3개의 NW 특징점들), 북쪽 특징점 유형(도 4에서의 4개의 N 특징점들), 북동쪽 특징점(도 4에서의 3개의 NE 특징점들), 동쪽 특징점(도 4에서의 4개의 E 특징점들), 남동쪽 특징점(도 4에서의 3개의 SE 특징점들), 남쪽 특징점(도 4에서의 4개의 S 특징점들), 남서쪽 특징점(도 4에서의 3개의 SW 특징점들), 서쪽 특징점(도 4에서의 4개의 W 특징점들), 및 중간 특징점(도 4에서의 5개의 M 특징점들)을 포함한다.

[0042] 전형적인 형태는 추출될 수 있는 9개의 유형들의 교차점들을 가진다. 다양한 유형들의 교차점들이 각각의 이전

에 추출된 선분들의 시작점 좌표 및 종점 좌표를 사용하여 추출된다.

[0043] 몇몇 실시예들에서, 9개의 유형들의 특징점을 추출하는 것 외에, 서버는 수량 정보와 같은 수평 및 수직 직선들(다시 말해서, 형태 경계들)에 관련된 정보를 추출한다. 다시 말해서, 서버는 수평 및 수직 직선들의 수를 추출한다.

[0044] 따라서, 서버는 총 11개의 유형들의 특징들을 추출한다. 서버는 형태의 구조적 특징들을 기술하는 특징 벡터들로서 이들 특징들을 출력하며, 여기에서 특징들은 특징점들 및 수평 및 수직 직선들과 관련된다. 예를 들면, 일 예로서 도 17을 사용하여, 각각의 차원(N, NE, E, SE, S, SW, W, NW, M, H, V)의 발생은 형태에 대해 11-차원 특징 벡터(예를 들면, (10, 1, 6, 1, 10, 1, 6, 1, 13, 8, 12))를 제공하기 위해 통산된다.

[0045] 특징점들은 좌표들이 아니다. 대신에, 특징점들은 다양한 종류들의 경계 특징들에 대한 통계적 값들이다. 직선 검출을 겪는 형태 이미지들에서, 픽셀들은 다음의 11개의 상태들 중 하나를 가진다: 북서쪽 픽셀, 북쪽 픽셀, 북동쪽 픽셀, 동쪽 픽셀, 남동쪽 픽셀, 남쪽 픽셀, 남서쪽 픽셀, 서쪽 픽셀, 중간 픽셀, 수평 선 및 수직 선. 선들 및 픽셀들의 좌표 값들은 요구되지 않는다. 형태 이미지에서의 형태의 구조적 특징들을 획득하기 위해, 11개의 유형들의 각각에 속하는 특징들의 수들이 카운팅된다.

[0046] 도 2로 가면, 240에서, 서버는 형태의 특징 벡터 및 다양한 템플릿 형태들의 특징 벡터들에 대한 가장 가까운 거리 대 두 번째로 가까운 거리의 비를 산출하며, 상기 산출에 기초하여 상기 형태를 식별한다. 다양한 템플릿 형태들은 상이한 수의 로우들, 상이한 수의 컬럼들, 간격 차이들, 포맷팅 차이들, 또는 그것의 임의의 조합을 가진다.

[0047] 동작(240)에서, "가장 가까운 거리"는 형태의 특징 벡터로부터 다양한 템플릿 형태들의 특징 벡터까지의 거리들 모두 중에서 최소 값을 가진 거리를 나타낸다. 가장 가까운 거리에 대응하는 템플릿 형태는 템플릿 형태들 모두 중에서, 식별될 형태에 가장 유사한 템플릿 형태이다. "두 번째로 가까운 거리"는 모든 거리 값들 중에서 두 번째로 작은 값을 갖는 것을 나타낸다. 단지 "가장 가까운 거리"만이 "두 번째로 가까운 거리"보다 작다.

[0048] 가장 가까운 거리 대 두 번째로 가까운 거리의 비는 설정 비보다 크거나 또는 같으며 가장 가까운 거리에 대응하는 형태 및 템플릿 형태 사이에서의 유사성이 가장 높은 것으로서 확인되는 경우에, 형태는 템플릿 형태 카테고리에 속하는 것으로 고려된다. 템플릿 형태 카테고리는 템플릿 형태에 의해 표현된 형태의 유형을 나타내며, 또한 템플릿 형태일 수 있다. 가장 가까운 거리 대 두 번째로 가까운 거리의 비가 설정 값보다 작은 경우에, 형태는 다양한 템플릿 형태들의 모두와 유사하지 않는 것으로 확인된다.

[0049] 도 5 및 도 6은 거리 유사성을 통해 식별되는 형태들의 실시예들의 다이어그램들이다.

[0050] 형태를 식별하기 위해 가장 가까운 거리 대 두 번째로 가까운 거리의 비를 사용하기 위한 많은 이유들이 존재한다. 예를 들면, 서버가 이미 "가장 가까운 거리"를 추출하였다고 가정하면, 입력 형태가 카테고리에 속하는지를 결정하기 위해 다양한 방법들이 존재한다. 입력 형태가 대응하는 템플릿 형태를 갖지 않을지라도, 가장 가까운 거리 값이 입력 형태를 위해 계산될 수 있다. 입력 형태가 카테고리에 속하는지를 결정하기 위한 하나의 방법은 임계 값(s)을 설정하는 것을 포함한다. 가장 가까운 거리가 s보다 작은 경우에, 입력 형태는 상기 카테고리의 형태들에 속한다. 가장 가까운 거리가 s보다 작지 않다면, 입력 형태는 형태 라이브리리에서의 임의의 카테고리에 속하지 않는다. 이러한 방법의 결점은 그것이 명백한 s 값을 찾아내는 것이 다소 어렵다는 것이다. 이러한 s 값은 많은 수동적 테스팅 후에만 결정될 수 있다. 또 다른 방법은 또한 두 번째로 가까운 거리를 찾아내며 가장 가까운 거리 및 두 번째로 가까운 거리 사이에서의 비를 산출하면서 임계 값(s)을 설정하는 것이다. 비가 s보다 큰 경우에, 형태는 "가장 가까운 거리" 카테고리에 속하는 것으로 고려되며, 비가 s보다 크지 않은 경우에, 형태는 도 5 및 도 6에 도시된 바와 같이, 형태 라이브리리에서의 임의의 카테고리에 속하지 않는다. 가장 가까운 거리 및 두 번째로 가까운 거리 사이에서의 큰 차이가 존재하는 경우에, 입력 형태는 S0 카테고리에 속할 가능성이 매우 높다. 도 5에 도시된 바와 같이, T0 및 T1(각각 가장 가까운 및 두 번째로 가까운 거리들을 나타내는) 사이에서의 차이는 매우 크며, 입력 형태는 S0 카테고리에 속한다. 가장 가까운 거리 및 두 번째로 가까운 거리 사이에 최소 차이가 존재하는 경우에, 입력 형태는 샘플 세트에서의 임의의 카테고리에 속할 가능성이 적다. 도 6에 도시된 바와 같이, T0 및 T1 사이에서의 차이가 충분히 크지 않기 때문에, 입력 형태는 샘플 세트에서의 임의의 카테고리에 속할 가능성이 적다. 따라서, 서버는 형태가 속하는 카테고리를 효과적으로 식별한다.

[0051] 형태가 카테고리에 속하는지를 결정하기 위해, 두 개의 형태들을 나타내는 두 개의 벡터들 사이에서의 유클리드 거리(Euclidian distance)가 산출된다. 예를 들면, 두 개의 동일한 형태들은 0의 거리를 가진다. 두 개의 형태

들 사이에서의 차이가 클수록, 산출된 거리는 더 크다.

[0052] 두 개의 형태들이 유사한지를 결정하기 위해, 제 1 방식은 임계치를 설정하는 것이다. 거리가 임계치보다 작으면, 두 개의 형태들은 유사하다. 그러나, 상이한 형태들은 상이한 임계치들을 가진다(예를 들면, 복잡한 형태들에 대한 임계치들은 간단한 형태들에 대한 임계치들과 상이하다).

[0053] 두 개의 형태들이 유사한지를 결정하는 제 2 방식은 상대적 임계치를 설정하는 것이다. 예를 들면, 상대적 임계치가 0.7이며 100개의 상이한 형태 템플릿들이 있다고 가정하자. 형태가 입력된 후, 탐색은 가장 가까운(N1) 형태 템플릿 및 다음으로 가까운(N2) 형태 템플릿에 대해 행해지며, T0 및 T1의 거리들이 각각 산출된다. T0 및 T1의 비는 N1이 상대적 임계치를 충족시키는 형태 템플릿이지를 결정하기 위해 사용된다. T0 및 T1이 가까우면, 입력 형태 및 N1은 매칭들이 아니며; 그렇지 않고, T0이 T1보다 상당히 더 작으면, 입력 형태는 N1과의 매칭이다.

[0054] 몇몇 실시예들에서, 서버는 형태 이미지 선분들에 대한 연결 구성요소 분석을 수행함으로써 형태 경계들을 획득한다. 상기 형태 이미지 선분들은 직선들이다. 통상적으로, 입력 형태는 단지 두 개의 유형들의 선들 또는 선분들을 가지며, 서버는 선분들 사이에서 구별한다. 서버는 형태 경계들을 보다 정확하고 효과적으로 만들기 위해 형태 경계들이 선 폭 제거 처리 및 문자 선분 필터링을 겪게 한다. 형태의 구조적 특징들은 형태 특징점들을 추출함으로써 획득된다. 그 뒤에, 서버는 유사성 분석에 기초하여 형태에 가장 유사한 템플릿 형태를 결정하며 그에 따라 형태를 식별한다. 따라서, 서버는 이미지 파일들 또는 형태 이미지들에서의 형태들을 바르게 식별하며, 컴퓨터들로의 그것의 편리한 및 빠른 입력을 위해 이미지 파일들에서의 형태들 또는 형태 이미지들에서의 형태들을 정확하게 식별한다.

[0055] 도 7은 형태 직선 검출 프로세스의 실시예의 흐름도이다. 몇몇 실시예들에서, 프로세스(700)는 도 2의 동작(210)의 구현이며, 다음을 포함한다:

[0056] 프로세스(700)는 본 출원의 형태 직선 검출 및 형태 카테고리 인식 양상들을 설명한다. 형태 직선 검출 양상에서, 상기 양상은 형태 경계 병합들에 관한 동작들을 포함한다. 형태 카테고리 인식 양상에서, 상기 양상은 형태들의 다중-분해능 프로세싱에 관한 동작들을 포함한다. 또한, 형태 모델들의 확립이 설명된다.

[0057] 먼저, 형태 직선 검출 양상은 형태 이미지를 수신하는 것, 상기 형태 이미지를 전-처리하는 것, 형태 이미지 연결 구성요소를 추출하는 것, 형태 이미지 선분들을 병합하는 것, 형태 이미지 문자 선분들을 필터링하는 것, 및 검출된 형태 이미지 직선들을 출력하는 것을 포함한다.

[0058] 710에서, 상기 서버는 입력된 형태 이미지를 수신한다.

[0059] 상기 입력 이미지 소스들은 스크린샷들, 사진 업로드들, 스캐너 업로드들, 또는 다른 방법들을 통해 획득된 이미지들을 포함한다. 입력 형태들의 카테고리들은 전자 형태들, 종이 노트들, 산업용 드로잉들 등을 포함한다.

[0060] 720에서, 서버는 형태 이미지를 전-처리한다.

[0061] 상기 전-처리는 입력된 형태 이미지의 그레이 스케일 프로세싱 및 이진 프로세싱을 포함한다. 몇몇 실시예들에서, 상기 전-처리된 출력 이미지는 단지 두 개의 컬러들, 흑색 및 백색을 포함한다.

[0062] 730에서, 서버는 형태 이미지로부터 연결 구성요소들을 추출한다.

[0063] 몇몇 실시예들에서, 서버는 연결 구성요소 방법에 기초하여 직선들을 추출하며, 이것은 비스듬한 직선들과 연관된 이슈들을 효과적으로 회피한다. 예를 들면, 형태를 스캐닝할 때, 선들은 일직선으로 스캐닝되지 않을 수 있으며 대신에 비스듬한 선들에 대해, 통상적인 수평/수직 스캔들이 적합하지 않다.

[0064] 도 8은 형태 이미지 프로세스로부터 연결 구성요소들의 추출의 실시예의 흐름도이다. 몇몇 실시예들에서, 프로세스(800)는 도 7의 동작(730)의 구현이며 다음을 포함한다:

[0065] 형태 경계들은 통상적으로 실질적으로 수평 또는 수직인 구조들을 보이기 때문에, 프로세스(800)가 사용된다.

[0066] 일 예로서, 프로세스(800)는 수평 직선들을 검출한다.

[0067] 810에서, 서버는 직선상에 하나의 점으로서 처리하기 위해 수직으로 배향된 연결 픽셀들을 수집한다. 몇몇 실시예들에서, 직선은 상이한 선 폭들과 호환 가능하다.

[0068] 몇몇 프로세싱된 형태 이미지들이, 흐릿함, 거칠기, 및 다른 이러한 상태들을 갖는, 열악한 품질을 갖는다는 사

실을 고려할 때, 각각의 픽셀에 수직으로 연결되는 픽셀들은 수평 직선 검출 동안 직선의 요소들로서 간주된다.

[0069] 820에서, 수평 방향(좌측 또는 우측면)으로 현재 픽셀에 인접한 픽셀이 존재하는 경우에, 서버는 현재 픽셀을 연결 영역에 부가한다.

[0070] 830에서, 서버는 추출된 연결 구성요소의 평균 폭(다시 말해서, 수평 선 상에서의 수직 픽셀들의 평균 수)을 산출한다. 평균 폭이 N이라고 가정하자. 영역에서의 각각의 수평 좌표에 대해, 현재 위치에서의 선 폭(수직 픽셀들의 수) 대 N의 비가 임계 값에 초과하는 경우에, 서버는 연결 구성요소로부터 현재 위치를 제거한다. 현재 위치가 직선 내에 있는 경우에, 현재 위치의 제거는 직선을 두 개의 세그먼트들로 분할한다. 이러한 동작은 잡음, 문자들, 및 수직 선들의 효과들을 감소시킨다.

[0071] 수직 직선 검출 방법은 수평 직선 검출 방법과 유사하며 유사한 방식으로 구현된다.

[0072] 다시 도 7을 참조하면, 740에서, 서버는 형태 이미지 선분들을 병합한다.

[0073] 통상적으로, 문자 선분들은 길이가 짧으며, 동일한 길이를 가진 문자 선분들의 빈도는 많은 문자 선분들이 유사한 길이를 갖기 때문에 높다. 도 13에서, 히스토그램은 문자 선분들의 비교적 집중된 분포를 반영하며, 통상적으로 문자 선분들은 짧은 길이를 가진 영역에 나타난다.

[0074] 형태 이미지 선분들의 병합은 원래 단일 선이지만 몇몇 이유로 끊어진 것으로 간주되는 다수의 직선들을 병합하는 것을 포함한다. 병합 동작 동안, 규칙 기반 결정은 동일한 방향으로 연장되는 두 개의 이웃하는 형태 경계들이 선분으로 병합될지를 결정하기 위해 이루어진다. 두 개의 이웃하는 형태 경계들이 병합된다면, 두 개의 이웃하는 형태 경계들은 선분 병합을 겪는다. 몇몇 실시예들에서, A) 서버는 동일한 방향으로 연장된 두 개의 이웃하는 형태 경계들의 거리를 획득하고, 상기 거리를 설정된 제 1 임계 값에 비교하며, 상기 설정된 제 1 임계 값에 대한 상기 거리의 비교에 기초하여 선분 병합을 수행할지를 결정하고, B) 서버가 동일한 방향으로 연장된 두 개의 이웃하는 형태 경계들의 유사성을 획득하며 상기 유사성에 기초하여 선분 병합을 수행할지를 결정하거나, 또는 C) 그것의 조합이다. 제 1 임계 값의 일 예는 12개의 픽셀들이다. 몇몇 실시예들에서, A) 서버는 동일한 방향으로 연장된 두 개의 이웃하는 형태 경계들의 유사성을 획득하며 상기 유사성에 기초하여 선분 병합이 수행될지를 결정하고, B) 서버는 동일한 방향으로 연장된 두 개의 이웃하는 형태 경계들 사이에서의 각도를 획득하고, 상기 각도를 설정된 제 2 임계값에 비교하며, 상기 비교에 기초하여 선분 병합이 수행될지를 결정하고, C) 서버가 동일한 방향으로 연장된 두 개의 이웃하는 형태 경계들 사이에서의 평행 거리를 획득하고, 상기 평행 거리를 설정된 제 3 임계 값에 비교하며, 상기 비교에 기초하여 선분 병합이 수행될지를 결정하거나, 또는 D) 그것의 임의의 조합이다. 제 2 임계 값의 일 예는 5 도들이다.

[0075] 다시 말해서, 두 개의 직선 세그먼트들의 병합은 이하의 조건들 중 적어도 하나를 충족시키도록 요구한다.

[0076] 조건 A: 두 개의 선분들(다시 말해서, 좌측면 선의 우측 끝 및 우측면 선의 좌측 끝) 사이에서의 가장 가까운 거리는 설정된 임계 값보다 작다. 몇몇 실시예들에서, 임계 값은 12 픽셀들로 설정된다. 거리가 임계 값보다 작은 경우에, 줄 바꿈(line break)은 정상적인 것으로 고려되며 두 개의 선분들이 병합된다. 일 양상으로서, 두 개의 선들이 원래 별개인 것으로 고려되며 병합되지 않는다.

[0077] 도 9a는 형태 직선 검출에서 선분 병합에서의 거리 결정의 실시예의 다이어그램이다. T1이 설정 임계 값보다 작을 때, 두 개의 선분들(좌측면 연결 구성요소(1-2-3-4-5-6-7-8) 및 우측면 연결 구성요소(1-2-3-4-5-6-7-8))이 병합된다. T1이 설정 임계 값보다 크거나 또는 동일할 때, 두 개의 선분들은 병합되지 않는다. 도 9b는 병합된 선분의 실시예의 다이어그램이다. 병합된 선분들은 병합된 선분을 형성하기 위해 채워진 캡을 가진다. 몇몇 실시예들에서, 캡은 1 픽셀의 폭을 가진 선으로 채워진다. 몇몇 실시예들에서, 캡은 1 이상의 픽셀의 폭을 가진 선으로 채워진다.

[0078] 조건 B: 두 개의 선분들의 유사성은 설정 임계 값보다 작다. 두 개의 선분들의, 유사성, 다시 말해서 선분들이 동일한 선일 가능성이 산출된다.

[0079] 유사성 산출은: 각각의 직선의 중심 픽셀 위치를 산출하고, 그 후 각각의 직선의 근사 표현을 획득하기 위해 선 맞춤 방법(예를 들면, 최소 제곱법)을 이용한다. 두 개의 선분들이 동일한 선일 가능성을 가중시키기 위해 두 개의 측정치들이 사용된다.

[0080] 제 1 측정치는 두 개의 선분들 사이에서의 각도(코사인 정리(cosine theorem)를 통해 획득될 수 있는)를 결정하는 것에 기초한다. 도 10은 형태 직선 검출에서 선분 병합에서의 각도 결정의 실시예의 다이어그램이다. 서버는 임의의 두 개의 추출된 연결 구성요소들의 수직 중심점들을 산출하며 상기 중심점들에 기초하여 가장 가까운 직

선을 맞춘다. 두 개의 연결 구성요소들 사이에서 각도는 기하학적 구조에 기초하여 산출되며, 따라서 연결 구성요소들 사이에서 각도의 산출을 성취한다. 각도가 작을수록, 두 개의 선들이 동일한 직선에 속할 가능성은 더 커진다.

[0081] 제 2 측정치는 두 개의 연결 구성요소들 사이에서의 평행 거리를 결정하는 것에 기초한다. 몇몇 실시예들에서, 평행 거리는 제 2 선분 픽셀 중심들로부터 제 1 선분까지의 점 선 거리의 제곱 평균의 합에 의해 표현된다. 도 11은 형태 직선 검출에서 선분 병합에서의 평행 선 거리 결정의 실시예의 다이어그램이다. 임의의 두 개의 추출된 연결 구성요소들의 경우에, 서버는 좌측면 연결 구성요소의 중심 위에 직선을 맞추며 그 후 우측면 연결 구성요소의 각각의 중심점에서 연장된 선까지의 거리들에 대한 제곱들의 합을 산출한다. 서버는 그에 의해 연결 구성요소들 사이에서의 평행 거리를 산출한다. 평행 거리가 작을수록, 두 개의 선들이 동일한 선일 가능성은 더 커진다. 몇몇 실시예들에서, 제 1 측정치에 대한 임계 값은 5 도이며, 제 2 측정치에 대한 임계 값은 제 2 연결 구성요소의 평균 선 폭 제곱들의 평균의 두 배이다. 다시 말해서, 두 개의 선분들은 두 개의 선분들이 이를 조건들 양쪽 모두를 충족시킬 때 단일 선으로 병합된다.

[0082] 선분 병합들은 통상적으로 복잡하거나 또는 열악한 이미지 품질을 가지며 "하나의 이유 또는 또 다른 이유로 끊긴 원래 단일 선들이었던 형태 선들"의 현상의 대상이 되는 몇몇 형태 이미지들에 대한 구제 조치이다. 선분 병합은 간단한, 선명한 형태 이미지들의 경우에 선택적이다.

[0083] 도 7을 다시 참조하면, 750에서, 서버는 형태 이미지 문자 선분들을 걸러낸다.

[0084] 몇몇 실시예들에서, 상기 선분 병합의 결과들은 문자 선분들을 포함할 수 있다. 통상적으로, 문자 선분들의 길이(예를 들면, 문자 획들에 대응하는)는 형태 경계에 비교하여 작으며 설정 임계 값을 갖고 걸러내어진다. 설정 임계 값의 일 예는 20 픽셀들이다.

[0085] 도 12는 형태 이미지 문자 선분 추출 프로세스의 실시예의 흐름도이다. 몇몇 실시예들에서, 프로세스(1200)는 동작(750)의 구현이다.

[0086] 프로세스(1200)는 상기 출력 결과들에 기초하여 선 길이들의 히스토그램을 생성하는 자동화된 임계 값 추출 프로세스를 포함한다.

[0087] 1210에서, 서버는 이전 출력 결과들에서 가장 긴 연결 영역의 길이를 획득한다.

[0088] 1220에서, 서버는 가장 긴 연결 영역을 M개의 동일한 간격들로 나눈다.

[0089] 1230에서, 각각의 선분에 대해, 서버는 각각의 선분의 길이에 따라 각각의 선분을 간격으로 배분하며, 이를 간격을 위한 카운터에 부가한다.

[0090] 1240에서, 서버는 배분된 선분들에 기초하여 선분들의 분포의 히스토그램을 생성한다. 서버는 문자 선분의 길이로서 작용하도록 최고점에 대응하는 길이를 선택한다. 따라서, 각각 수평 및 수직 방향들로 문자 폭들 및 문자 높이들을 획득하는 것이 가능하며, 서버는 선분들을 스크리닝하기 위한 임계 값들로서 문자 폭들 및 문자 높이들을 사용한다. 문자 폭들 및 문자 높이들은 사전 구성되지 않지만 히스토그램 분석에 기초하여 결정된다. 히스토그램의 피크에 대응하는 세그먼트 길이는 문자 선분 길이로서 사용된다.

[0091] 1250에서, 서버는 임계 값보다 높은 선분들만을 보유한다(다시 말해서, 임계 값의 우측에 대한 간격들로 배분된 선분들).

[0092] 도 13은 형태 직선 검출에서 문자 선분 필터링의 실시예의 다이어그램이다. 도 12의 상기 문자 선분 필터링 프로세스는 도 13을 고려하여 설명된다.

[0093] 일 예로서, 모든 길이들은 다수의 픽셀들로 표현된다. 가장 긴 연결 구성요소(검출된 가장 긴 직선에 의해 표현된)의 길이는 500이며, M=100이라고 가정하자. 다시 말해서, 전체 공간은 총 100개의 간격들(0, 5, 10, ..., 495, 500)로 한정된다. 서버는 전체 형태 이미지에서 검출된 직선들의 픽셀 길이들에 기초하여 히스토그램을 생성한다. x-좌표들은 픽셀 간격들이며, y-좌표들은 빈도들이다. "M=100"은 단지 예시적인 예로서 작용한다는 것을 주의하자. 실제 애플리케이션들에서, M의 값은 실제 조건들에 따라 설정될 수 있다. 화상이 큰 경우에, 많은 픽셀들이 존재할 것이다. 따라서, M은 적절하게 증가되거나 또는 감소된다.

[0094] 상기 직선 검출은 복수의 직선 세그먼트들(실제 형태 경계들 및 문자 획 선분들을 포함한)을 출력한다. 거의 모든 상황들에서, 형태 경계들은 문자 획 선분들보다 더 길며 더 적다(특히, 밀집한 텍스트가 포함되는 경우에). 그러므로, 이러한 동작은 피크 값을 고려할 때(도 13에서의 원형 부분에 대응하는 문자 선분의 길이) 임계 값을

검출하며 임계값보다 큰 선분들(실제 형태 경계들)만을 보유한다.

[0095] 몇몇 실시예들에서, 문자 선분 필터링은 훈련을 요구한다. 훈련 또는 인식 프로세스 동안, 입력 형태 이미지는 다양한 텍스트 입력 콘텐트를 포함한다. 형태로 입력된 문자들의 양이 설정 값보다 큰 경우에, 입력 문자들은 형태 경계들에 대하여 보다 작은 직선들을 생성하며, 이것은 후속 동작들에 영향을 미칠 것이다. 정확도를 증가시키기 위해 문자 선분들을 걸러내며 형태 인식 상에서의 텍스트 입력 콘텐트의 영향을 감소시킨다.

[0096] 도 7로 가면, 760에서, 서버는 형태 이미지 직선 검출 결과들을 출력한다.

[0097] 또한, 각각의 선분에 대해, 서버는 Pstart: 시작점 좌표; Pend: 종점 좌표; L: 길이; 및 W: 평균 폭을 보유한다.

[0098] 형태상에서 직선 검출을 수행한 후, 서버는 형태 카테고리 인식을 수행한다.

[0099] 도 14는 형태 카테고리 인식 프로세스의 실시예의 흐름도이다. 프로세스(1400)는 도 7의 프로세스(700) 후 수행되며 다음을 포함한다.

[0100] 1410에서, 서버는 형태 특징들을 추출한다.

[0101] 상이한 유형들의 형태들은 상이한 구조들을 가진다. 몇몇 실시예들에서, 일반적인 접근법은 형태들을 설명하고, 형태 구조적 특징들을 획득하며, 형태 구조적 특징들에 기초하여 형태 카테고리들을 결정하기 위해 사용된다.

[0102] 도 15는 형태 특징들 프로세스의 추출의 실시예의 흐름도이다. 몇몇 실시예들에서, 프로세스(1500)는 동작(1410)의 구현이며 다음을 포함한다:

[0103] 1510에서, 서버는 교차점(특징점)을 추출한다. 공통 형태는 추출될 수 있는 총 9개 유형들의 교차점을 가진다. 각각의 이전에 추출된 선분은 적어도 시작점 좌표 및 종점 좌표를 포함하며, 이것은 특징점을 추출하기 위해 사용될 수 있다. 도 4를 참조하면, 몇몇 실시예들에서, 특징점 추출은 추출되는 9개 유형들의 특징점을 포함한 형태 특성들에 기초한다: 북서쪽 특징점, 북쪽 특징점, 북동쪽 특징점, 동쪽 특징점, 남동쪽 특징점, 서쪽 특징점, 남서쪽 특징점, 서쪽 특징점, 및 중간 특징점.

[0104] 도 16은 형태 카테고리 인식을 겪는 다중-레벨, 계층화된 형태의 다이어그램이다. 도 17은 형태상에서 수행된 직선 검출을 따르는 출력 결과들의 차트이다. 도 18은 형태 출력 결과상에서 수행된 특징점 추출에 따른 특징점들의 다이어그램이다.

[0105] 일 예로서 도 16에서의 형태를 사용하여, 형태가 직선 검출을 겪은 후의 출력 형태가 도 17에 도시된다. 도 17에 묘사된 형태는 도 18에 도시된 바와 같이, 교차점 추출을 겪으며, 여기에서 NW는 북서쪽 특징점을 나타내고, N은 북쪽 특징점을 나타내고, NE는 북동쪽 특징점을 나타내고, E는 동쪽 특징점을 나타내고, SE는 남동쪽 특징점을 나타내고, S는 서쪽 특징점을 나타내고, SW는 남서쪽 특징점을 나타내고, W는 서쪽 특징점을 나타내며, M은 중간 특징점을 나타낸다. 또한, 수평 선들의 수(H=8) 및 수직 선들의 수(V=12)가 존재한다. 각각의 차원(N, NE, E, SE, S, SW, W, NW, M, H, V)의 발생들은 이들 차원들에 기초하여 이러한 형태에 대한 11-차원(10, 1, 6, 1, 10, 1, 6, 1, 13, 8, 12) 특징 벡터를 제공하도록 통산된다.

[0106] 다중-분해능 프로세싱은 증가된 정확도를 달성하기 위해 형태상에서 수행된다. 다중-분해능 프로세싱 하에서, 상기 형태의 특징 벡터는 상이한 서브-윈도우들 하에서 상기 차원 특징들의 각각의 총합이다.

[0107] 도 15로 가면, 1520에서, 서버는 형태의 외부 프레임 영역을 결정한다. 형태의 외부 프레임 영역은 상기 추출된 4개의 코너 특징점들에 기초하여 결정된다.

[0108] 1530에서, 서버는 다중-분해능 프로세싱을 수행한다. 형태가 위치되는 영역을 획득한 후, 제 1 단계는 형태의 크기를 정규화하기 위한 것이며, 예를 들면, 형태를 640×480 으로 균일하게 수축 또는 팽창시킨다. 서버는 정규화된 이미지를 $N \times N$ 블록들로 나눈다. 도 16에 도시된 바와 같이, 원래 이미지는 3개의 레벨들로 나뉜다.

[0109] 제 1 레벨에서, 11 차원들이 원래 이미지로부터 추출된다. 다음으로, 제 2 레벨에서, 원래 이미지는 $2 \times 2 = 4$ 서브 영역들로 나뉘며, 따라서 $4 \times 11 = 44$ 차원들이 있다. 제 3 레벨에서, 원래 이미지는 $4 \times 4 = 16$ 서브영역들로 나뉘며, 따라서 $16 \times 11 = 176$ 차원들이 있다. 제 1, 제 2, 및 제 3 레벨들의 차원들은 차원들의 총 수가 = $11 + 44 + 176 = 231$ 이도록 합산된다.

[0110] 1540에서, 서버는 특징 추출을 수행한다. 총 11개 유형들의 특징들이 본 예를 갖고 추출된다. 11개의 유형들의 특징들은 동작(1520)에서 추출된 9개의 특징 수량들, 뿐만 아니라 수평 및 수직 선들을 포함한다. 다시 말해서,

서버는 형태의 각각의 블록 내에서 다양한 형태 경계들 및 상기 다양한 형태 경계들 사이에서의 위치 관계들에 기초하여 다중-블록 형태에서의 형태의 각각의 블록으로부터 교차점들을 추출하고, 상기 형태의 각각의 블록에서의 형태 경계들의 다양한 교차점들을 추출하고, 상기 형태의 각각의 블록에서의 상기 다양한 교차점들에 대한 정보를 획득하며, 상기 형태의 각각의 블록에서 수평으로 배향된 형태 경계들에 대한 정보 및 수직으로 배향된 형태 경계들에 대한 정보를 획득한다.

- [0111] 1550에서, 서버는 특징 히스토그램을 형성한다. 다양한 이미지 서브-윈도우들에 대한 특징 수량들은 도 16의 다중-레벨, 계층화된 이미지로부터 추출된다. 특징 수량들은 그 후 특징 히스토그램을 형성하기 위해 병합된다. 도 16에서의 3-층 이미지에 대해, 총 $(1+4+16) \times 11 = 231$ 차원들을 가진 특징들이 추출된다.
- [0112] 1560에서, 서버는 특징들을 출력한다. 상기 231 차원 특징들은 형태의 구조를 설명하기 위한 특징 벡터 출력으로서 간주된다.
- [0113] 상기로부터, 형태 인식을 위한 11-차원 벡터의 사용은 다수의 예를 들면, 충분하지 않은 치수들, 불충분한 정보 등)을 야기할 수 있지만, 인식 정확도가 다중-분해능 프로세싱을 통해 증가된다. 예를 들면, 텍스트에서의 3개의 분해능들($1, 2 \times 2, 4 \times 4$)을 사용함으로써, 최종 결과는 결과적으로 증가된 인식 정확도를 위해 $11 \times 21 = 231$ 차원들이 된다.
- [0114] 1420에서, 서버는 다양한 모델들(템플릿 형태들)에 대한 유사성들을 산출하며 형태가 속하는 형태 카테고리를 결정한다.
- [0115] 예를 들면, 입력 형태의 특징 벡터로부터 다양한 템플릿 형태들의 특징 벡터들까지의 유클리드 거리들이 산출된다. 형태 템플릿 라이브러리에 속하는 상기 입력 형태가 알려져 있는 경우에, 서버는 최단 유클리드 거리를 가진 형태 유형을 직접 리턴한다. 그렇지 않으면, 서버는 다양한 모델들에 대한 형태의 유사성들을 산출한다.
- [0116] 도 19는 다양한 모델들 프로세스에 대한 형태의 유사성들의 산출의 실시예의 흐름도이다. 몇몇 실시예들에서, 프로세스(1900)는 동작(1420)의 구현이며 다음을 포함한다:
- [0117] 1910에서, 서버는 입력 형태 이미지에서의 형태 및 다양한 템플릿 형태들로부터 가장 가까운 유클리드 거리 및 두 번째로 가까운 유클리드 거리 사이에서의 비를 산출한다.
- [0118] 1920에서, 서버는 비가 T보다 큰지 또는 동일한지를 결정한다.
- [0119] 1930에서, 비가 T보다 크거나 또는 동일한 경우에, 서버는 입력 형태 이미지에 가장 가까운 템플릿 형태를 되돌려 보낸다.
- [0120] 1940에서, 비가 T보다 작은 경우에, 서버는 입력 형태 이미지가 임의의 템플릿 형태와 매칭되지 않음을 결정한다. 몇몇 실시예들에서, T는 3으로 설정된다.
- [0121] 도 20은 형태 모델 프로세스의 확립의 실시예의 흐름도이다. 몇몇 실시예들에서, 프로세스(2000)는 도 14의 동작(1410) 후에 구현되며 다음을 포함한다:
- [0122] 이 예에서, 형태 모델(즉, 템플릿 형태)은 프로세스(2000)를 통해 셋업된다. 프로세스(2000)는 샘플 화상(형태 템플릿) 입력, 샘플 화상 직선 검출 및 추출, 샘플 화상 특징 추출, 및 형태 모델 확립을 포함한다.
- [0123] 몇몇 실시예들에서, 샘플 화상(형태 템플릿) 입력 및 샘플 화상 직선 검출 및 추출은 형태 직선 검출 프로세스(700)의 관련 섹션들에 대한 참조를 계속한다. 샘플 화상 특징 추출에 대하여, 특징 추출은 동작(1410)에 대한 참조를 계속할 수 있다.
- [0124] 2010에서, 샘플 화상에서의 형태에 대한 특징 추출 및 특징 벡터들의 추출이 수행된 후, 서버는 형태의 카테고리를 설명하며 그에 따라 템플릿 형태를 생성하는 모델 출력으로서 이들 특징 벡터들을 간주한다.
- [0125] 예를 들면, 상이한 형태 템플릿들(송장 A, 형태 B, 변상 형태 C와 같은)이 준비된다. 시스템은 상이한 형태 템플릿들이 직선 검출 및 특징 추출을 겪게 한다. 최종 훈련 출력 결과는 각각의 형태 모델이 하위에 도시된 바와 같이 다-차원 벡터(예를 들면, 231 차원들)에 대응한다는 것이다: A: {231-차원 벡터}, B: {231-차원 벡터}, 및 C: {231-차원 벡터}.
- [0126] 형태 화상 구조들을 위한 자동 인식 프로세스가 제공된다. 상기 자동 인식 프로세스는 형태 화상들의 직선 검출에서의 밝기, 기울기, 끊김들, 잡음, 및 상이한 선분 폭들에 강력하게 대응한다. 몇몇 실시예들에서, 입력 형태 이미지들은 스크린샷들, 사진, 스캐너 업로드들, 또는 다른 방법들을 통해 획득된다. 몇몇 실시예들에서, 이

미지 소스들은 전자 형태들 또는 손으로 그린 차트들이다. 게다가, 화상 품질에 대한 어떤 요건도 존재하지 않는다. 샘플 라이브러리가 많은 템플릿 형태들을 포함할 때, 방법은 훈련을 통해, 개개의 입력 형태들이 속하는 카테고리들을 자동으로 인식하는 형태 특징들을 설명하기 위해 존재한다. 게다가, 훈련 프로세스를 제외하면, 전체 인식 프로세스는 임의의 수동적 간섭 없이 자동화될 수 있다.

[0127] 도 21은 형태 인식 디바이스의 실시예의 구조적 블록 차트이다. 몇몇 실시예들에서, 디바이스(2100)는 도 22의 서버(2220)의 구현이며 직선 검출 모듈(2110), 특징 추출 모듈(2120), 유사성 산출 모듈(2130), 및 형태 인식 모듈(2140)을 포함한다.

[0128] 직선 검출 모듈(2110)은 형태 이진 이미지에서 형태의 직선 검출을 행하며 형태의 다양한 형태 경계들 및 상기 다양한 형태 경계들 사이에서의 위치 관계들을 획득한다.

[0129] 특징 추출 모듈(2120)은 형태로부터 특징을 추출하며 다양한 형태 경계들 및 상기 다양한 형태 경계들 사이에서의 위치 관계들에 기초하여 형태로부터 특징 벡터를 획득한다. 형태 벡터들은 형태의 구조적 특징들을 나타낸다.

[0130] 유사성 산출 모듈(2130)은 형태의 특징 벡터에 기초하여 형태 및 템플릿 형태들 사이에서의 유사성을 산출한다.

[0131] 형태 인식 모듈(2140)은 산출된 유사성들에 기초하여 형태를 식별한다.

[0132] 몇몇 실시예들에서, 직선 검출 모듈(2110)은 시작점으로서 형태상에서의 각각의 직선상에서 설정점을 사용하고; 각각의 직선상에서 인접한 픽셀들에 대한 연결 구성요소 분석을 행하고; 각각의 직선의 연결 구성요소들을 획득하고; 각각의 직선의 연결 구성요소들 상에서 각각의 픽셀에 대한 각각의 직선에 수직인 연결 구성요소들의 픽셀들의 수를 획득하고; 각각의 직선의 연결 구성요소들 상에서 각각의 픽셀에 대한 각각의 직선에 수직인 연결 구성요소들의 픽셀들의 수에 기초하여 각각의 직선의 평균 선 폭을 획득하고; 각각의 픽셀이 각각의 직선의 연결 구성요소들 상에 위치되는 선 폭이 평균 선 폭보다 큰지를 결정하고; 각각의 픽셀이 각각의 직선의 연결 구성요소들 상에 위치되는 선 폭이 평균 선 폭보다 큰 경우에, 평균 선 폭보다 큰 픽셀들이 위치되는 선 폭에 대한 선 폭 제거 처리를 수행하며; 상기 선 폭 제거 처리를 겪은 각각의 직선에 기초하여 형태의 다양한 형태 경계들 및 상기 다양한 형태 경계들 사이에서의 위치 관계들을 획득한다.

[0133] 몇몇 실시예들에서, 직선 검출 모듈(2110)이 평균 선 폭보다 큰 픽셀들이 위치되는 선 폭 상에서 선 폭 제거 처리를 수행할 때; 직선의 연결 구성요소들로부터 평균 선 폭보다 큰 픽셀들을 제거하고; 제거된 픽셀들의 위치들이 직선 내에 있는지를 결정하며; 제거된 픽셀들의 위치들이 직선 내에 있는 경우에, 제거된 픽셀들의 양쪽 측면들 상에서 선분들을 병합한다.

[0134] 몇몇 실시예들에서, 특징 추출 모듈(2120)은 다양한 형태 경계들 및 상기 다양한 형태 경계들 사이에서의 위치 관계에 기초하여 형태 경계들의 다양한 교차점들을 추출하고; 상기 다양한 교차점들에 대한 정보를 획득하고; 상기 다양한 형태 경계들 중에서 수평으로 배향된 형태 경계들에 대한 정보 및 수직으로 배향된 형태 경계들에 대한 정보를 획득하고; 상기 다양한 교차점들에 대한 정보, 상기 수평으로 배향된 형태 경계들에 대한 정보, 및 상기 수직으로 배향된 형태 경계들에 대한 정보에 기초하여 상기 형태의 특징 벡터를 획득한다.

[0135] 몇몇 실시예들에서, 상기 형태 인식 디바이스(2100)는 형태 프로세싱 모듈(2150)을 더 포함한다.

[0136] 형태 프로세싱 모듈(2150)은 형태 경계들의 다양한 교차점들의 추출 이전에, 상기 형태의 영역을 결정하며, 상기 형태의 영역에 기초하여 상기 형태에 대한 정규화 처리를 수행한다.

[0137] 몇몇 실시예들에서, 상기 형태 프로세싱 모듈(2150)은 형태가 정규화된 후 상기 형태를 복수의 블록들로 나누며, 상기 특징 추출 모듈(2120)은 상기 형태의 각각의 블록 내에서 상기 다양한 형태 경계들 및 상기 다양한 형태 경계들 사이에서의 위치 관계들에 기초하여 상기 복수의 블록들 형태에서 상기 형태의 각각의 블록으로부터 교차점들을 추출하고; 상기 형태의 블록에서 상기 형태 경계들의 다양한 교차점들을 추출하고; 상기 형태의 블록에서 상기 다양한 교차점들에 대한 정보를 획득하고; 상기 형태의 블록에서 수평으로 배향된 형태 경계들에 대한 정보 및 수직으로 배향된 형태 경계들에 대한 정보를 획득하며; 각각의 형태 블록에서 상기 다양한 교차점들에 대한 정보, 상기 수평으로 배향된 형태 경계들에 대한 정보, 및 상기 수직으로 배향된 형태 경계들에 대한 정보에 기초하여 상기 형태의 특징 벡터를 획득한다.

[0138] 몇몇 실시예들에서, 유사성 산출 모듈(2130)은 형태의 특징 벡터 및 다양한 템플릿 형태들의 특징 벡터들에 대한 가장 가까운 거리 대 두 번째로 가까운 거리의 비를 산출한다; 비가 설정 비보다 크거나 또는 동일한

경우에, 가장 가까운 거리에 대응하는 형태 및 템플릿 형태 사이에서의 유사성을 가장 높은 것으로 결정하며; 비가 설정 비보다 작은 경우에, 상기 형태를 다양한 템플릿 형태들의 각각과 유사하지 않은 것으로 결정한다.

[0139] 몇몇 실시예들에서, 형태 인식 디바이스(2100)는 선분 병합 모듈(2160)을 더 포함한다.

[0140] 형태상에서의 특징 추출의 수행 및 상기 형태로부터의 상기 특징 벡터의 획득 전에, 선분 병합 모듈(2160)은 동일한 방향으로 연장되어 인접한 두 개의 형태 경계들이 설정 규칙에 기초하여 선분 병합을 겪을지를 결정하며; 동일한 방향으로 연장되어 인접한 두 개의 형태 경계들이 선분 병합을 겪는 경우에, 두 개의 인접한 형태 경계들에 대한 선분 병합을 수행한다.

[0141] 몇몇 실시예들에서, 선분 병합 모듈(2160)은 동일한 방향으로 연장된 두 개의 인접한 형태 경계들 사이에서의 거리를 획득하며 설정된 제 1 임계 값에 거리를 비교한 결과에 기초하여 선분 병합이 수행될지를 결정하고, 설정된 제 1 임계 값에 거리를 비교한 결과에 기초하여 동일한 방향으로 연장된 두 개의 인접한 형태 경계들의 유사성을 획득하거나, 또는 그것의 조합을 행하고; 선분 병합이 수행될지를 결정하며; 선분 병합이 수행되는 경우에, 두 개의 인접한 형태 경계들을 병합한다.

[0142] 몇몇 실시예들에서, 선분 병합 모듈(2160)이 동일한 방향으로 연장되는 두 개의 인접한 형태 경계들의 유사성을 획득하며 상기 유사성에 기초하여 상기 선분 병합이 수행될지를 결정할 때, 선분 병합 모듈(2160)은 동일한 방향으로 연장된 두 개의 인접한 형태 경계들 사이에서의 각도를 획득하며 설정된 제 2 임계 값에 상기 각도를 비교한 결과에 기초하여 선분 병합이 수행될지를 결정하고; 동일한 방향으로 연장된 두 개의 인접한 형태 경계들 사이에서의 평행 거리를 획득하며 설정된 제 3 임계 값에 상기 평행 거리를 비교한 결과에 기초하여 선분 병합이 수행될지를 결정하거나; 또는 그것의 조합을 행한다.

[0143] 몇몇 실시예들에서, 형태 인식 디바이스(2100)는 문자 선분 필터링 모듈(2170)을 더 포함한다.

[0144] 특징 추출 모듈(2120)이 형태로부터 특징을 추출하며 다양한 형태 경계들 및 상기 다양한 형태 경계들 사이에서의 위치 관계들에 기초하여 상기 형태로부터 특징 벡터들을 획득하기 전에, 상기 문자 선분 필터링 모듈(2170)은 형태의 다양한 형태 경계들의 문자 선분 필터링을 수행한다.

[0145] 몇몇 실시예들에서, 문자 선분 필터링 모듈(2170)은 형태의 다양한 형태 경계들의 선분 길이들이 설정된 제 4 임계 값보다 작은지 또는 동일한지를 결정하며; 상기 형태의 다양한 형태 경계들의 선분 길이들이 상기 설정된 제 4 임계 값보다 작거나 또는 동일한 경우에, 그것의 선분 길이가 상기 제 4 임계 값보다 작거나 또는 동일한 형태 경계들을 제거한다.

[0146] 몇몇 실시예들에서, 제 4 임계 값은 다음과 같이 설정된다: 디바이스(2100)는 형태의 다양한 형태 경계들에서의 최대 연결 구성요소의 길이를 획득하며 상기 길이를 복수의 길이 간격들로 나누고; 길이에 기초한 다양한 형태 경계들에서의 각각의 연결 구성요소를 복수의 길이 간격들 중에서의 길이 간격으로 배분하고; 상기 복수의 길이 간격들의 길이들 및 상기 길이 간격들의 각각의 연결 구성요소들의 수에 기초하여 선분 길이 분포 히스토그램을 생성하며; 상기 히스토그램에서의 y-좌표의 최고점에 대응하는 길이를 제 4 임계 값으로 설정한다.

[0147] 형태 인식 디바이스(2100)는 상기 형태 인식 방법들 중 적어도 하나를 구현한다. 게다가, 형태 인식 디바이스(2100)는 대응하는 프로세스들의 유익한 결과들을 가진다. 따라서, 디바이스(2100)는 간결성을 위해 추가로 논의되지 않을 것이다.

[0148] 도 22는 형태 인식을 위한 시스템의 실시예의 구조도이다. 시스템(2200)은 네트워크(2230)를 통해 서버(2220)에 연결된 클라이언트(2210)를 포함한다.

[0149] 본 출원은 두 개의 주요 부분들을 포함하는 형태 이미지들에 대한 자동 인식 프로세스를 제공한다: 입력 형태 이미지들의 직선 검출 및 형태 카테고리 결정. 직선 검출에 관하여, 본 출원은 연결 구성요소 분석에 기초하여 직선들을 추출하기 위한 프로세스를 제공한다. 이러한 프로세스는 형태 경계들의 정확하고, 선명한 추출들을 가능하게 한다. 형태 카테고리 결정은 형태들에 대한 다중-분해능 특징 추출을 수행하기 위한 프로세스를 제공한다. 이러한 프로세스는 형태 인식의 정확도를 증가시킬 수 있다.

[0150] 상기 설명된 유닛들은 하나 이상의 범용 프로세서들 상에서 실행하는 소프트웨어 구성요소들로서, 특정한 기능들을 수행하도록 설계된 프로그램 가능한 논리 디바이스들 및/또는 애플리케이션 특정 집적 회로들과 같은 하드웨어 또는 그것의 조합으로서 구현될 수 있다. 몇몇 실시예들에서, 유닛들은 컴퓨터 디바이스(개인용 컴퓨터들, 서버들, 네트워크 장비 등과 같은)가 본 발명의 실시예들에 설명된 방법들을 구현하게 하기 위한 다수의 명령들을 포함하여, 비휘발성 저장 매체(광학 디스크, 플래시 저장 디바이스, 이동 하드 디스크 등과 같은)에 저장될

수 있는 소프트웨어 제품들의 형태에 의해 구체화될 수 있다. 유닛들은 단일 디바이스 상에 구현되거나 또는 다수의 디바이스들에 걸쳐 분배될 수 있다. 유닛들의 기능들은 서로 병합되거나 또는 다수의 서브-유닛들로 추가로 분리될 수 있다.

[0151] 여기에 개시된 실시예들을 고려하여 설명된 방법들 또는 알고리즘 단계들은 하드웨어, 프로세서-실행 소프트웨어 모듈들, 또는 양쪽 모두의 조합들을 사용하여 구현될 수 있다. 소프트웨어 모듈들은 랜덤-액세스 메모리(RAM), 메모리, 판독-전용 메모리(ROM), 전기적으로 프로그램 가능한 ROM, 전기적으로 소거 가능한 프로그램 가능한 ROM, 레지스터들, 하드 드라이브들, 착탈 가능한 디스크들, CD-ROM, 또는 기술 분야에 알려진 임의의 다른 형태들의 저장 미디어에 설치될 수 있다.

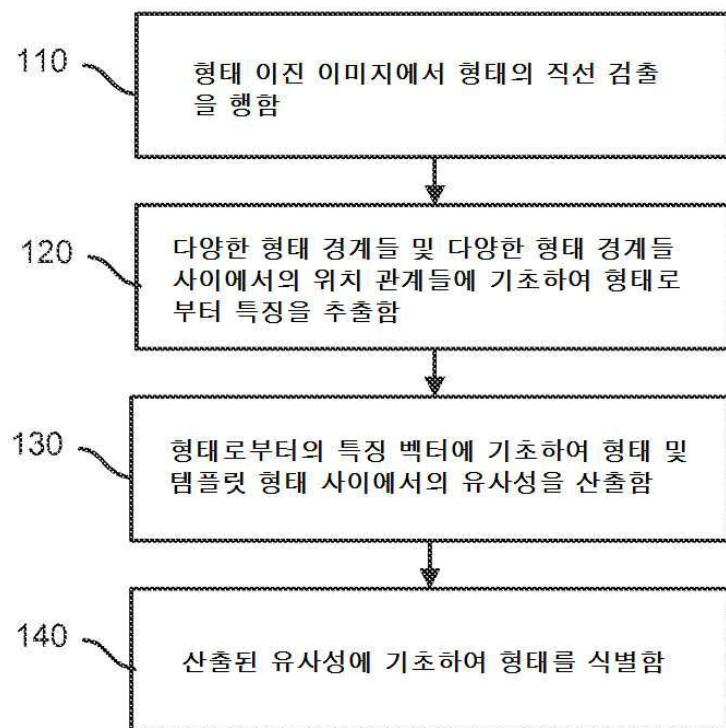
[0152] 앞서 말한 실시예들은 이해의 명료함을 위해 어느 정도 상세히 설명되었지만, 본 발명은 제공된 세부사항들에 제한되지 않는다. 본 발명을 구현하는 많은 대안적인 방식들이 있다. 개시된 실시예들은 예시적이며 제한적이지 않다.

부호의 설명

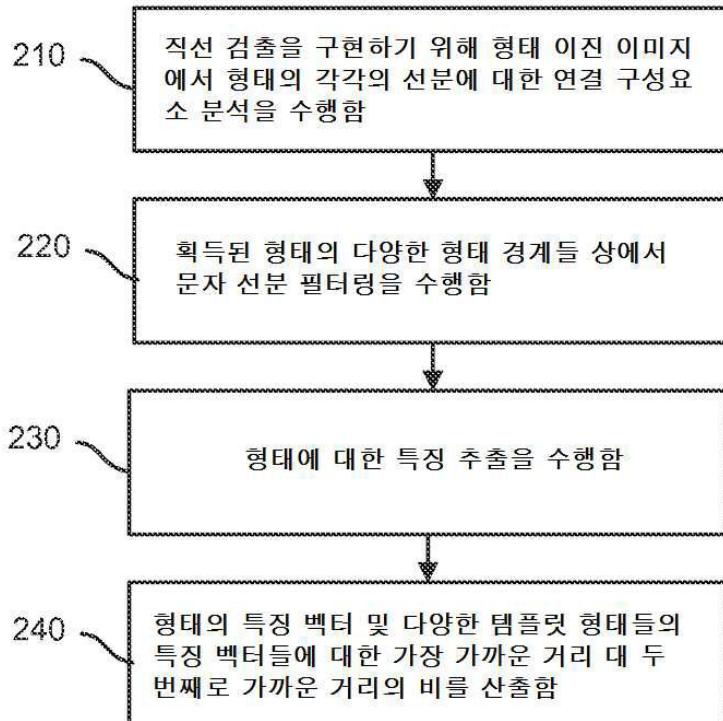
2100: 형태 인식 디바이스	2110: 직선 검출 모듈
2120: 특징 추출 모듈	2130: 유사성 산출 모듈
2140: 형태 인식 모듈	2150: 형태 프로세싱 모듈
2160: 선분 병합 모듈	2170: 문자 선분 필터링 모듈
2200: 형태 인식 시스템	2210: 클라이언트
2220: 서버	2230: 네트워크

도면

도면1

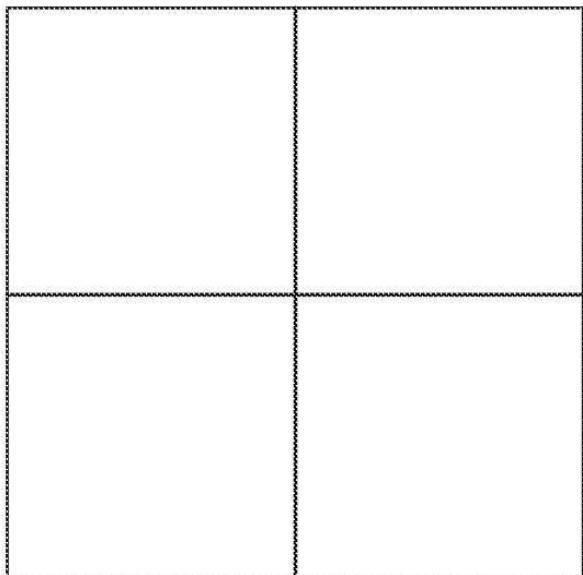


도면2



200

도면3a



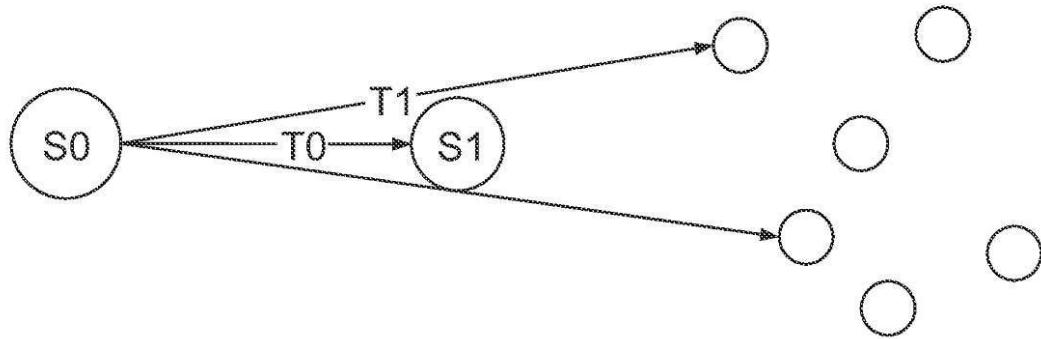
도면3b

			4				
	2		4	5		7	
1	2	3	4	5	6	7	8
1	2	3	4		6	7	
		3			6		

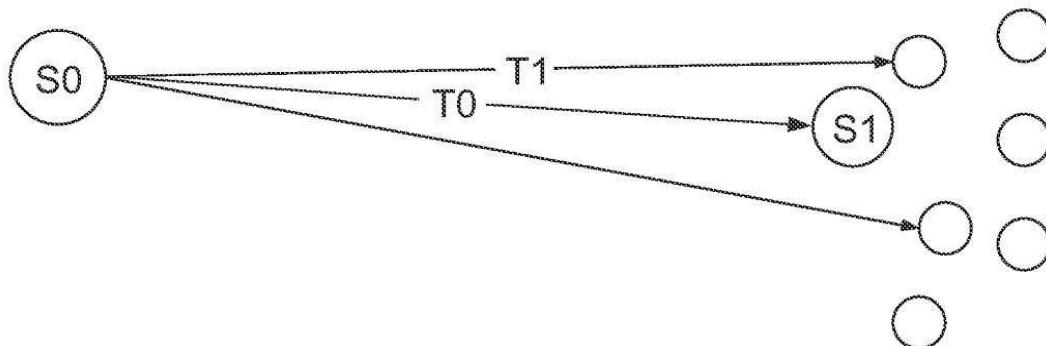
도면4

NW	NW	N	N	N	NE	NE
NW			N			NE
W			M			E
W	W	M	M	M	E	E
W			M			E
SW			S			SE
SW	SW	S	S	S	SE	SE

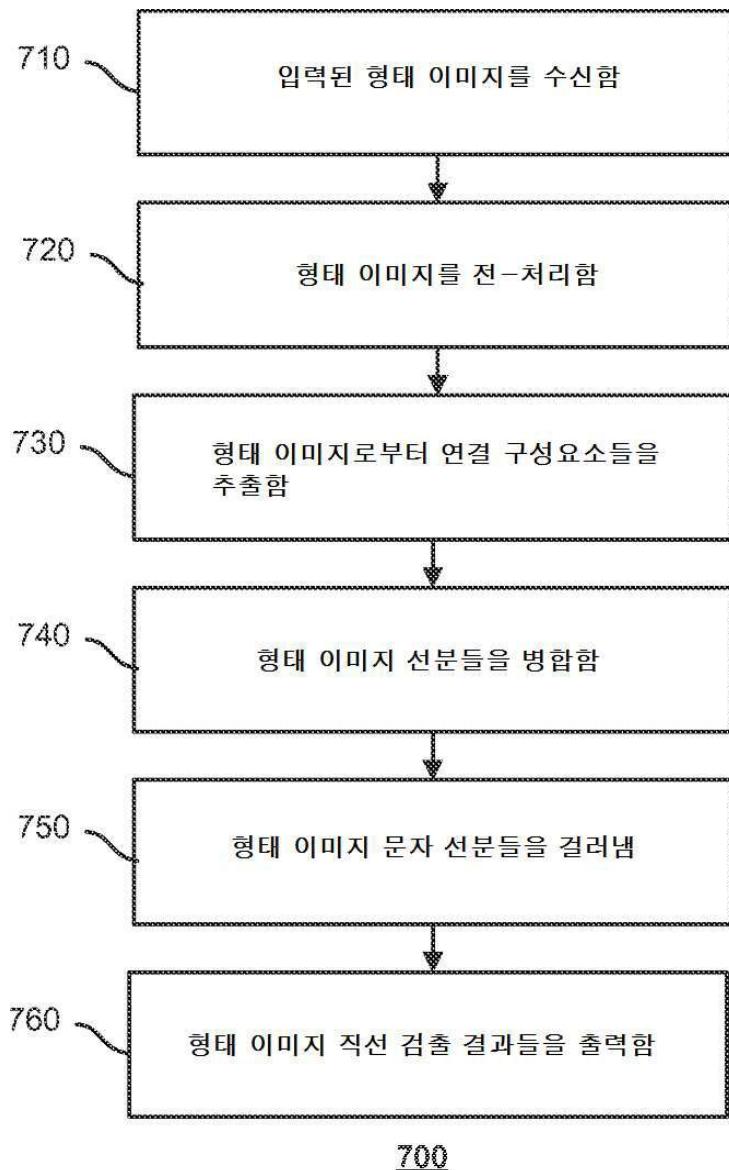
도면5



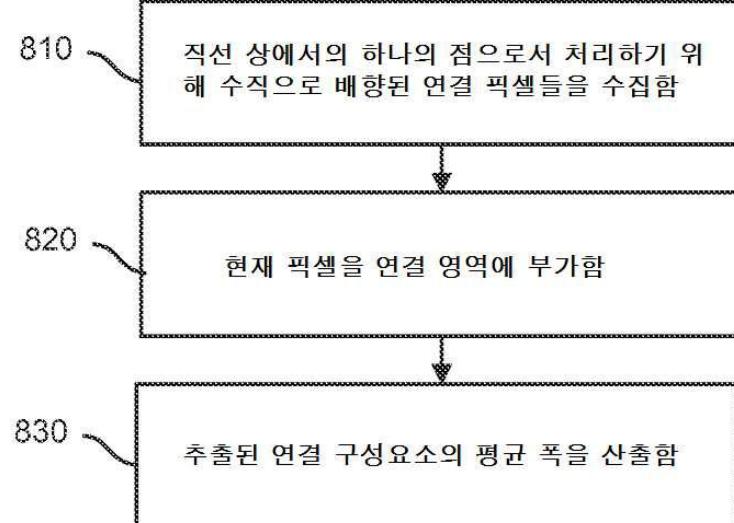
도면6



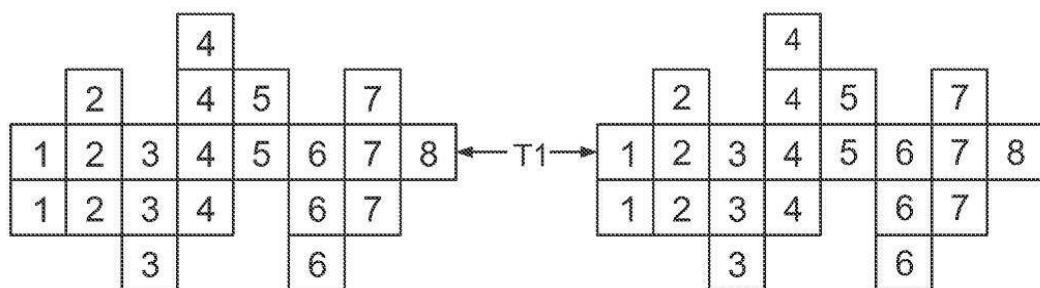
도면7



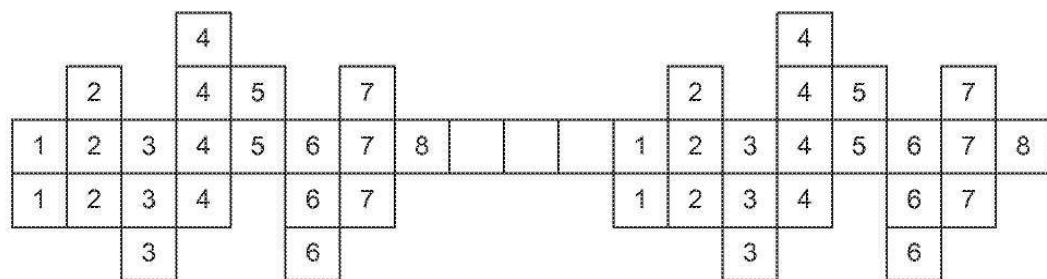
도면8

800

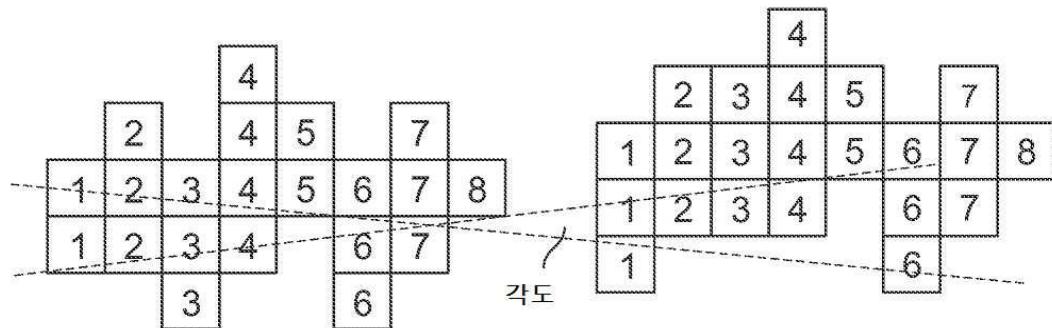
도면9a



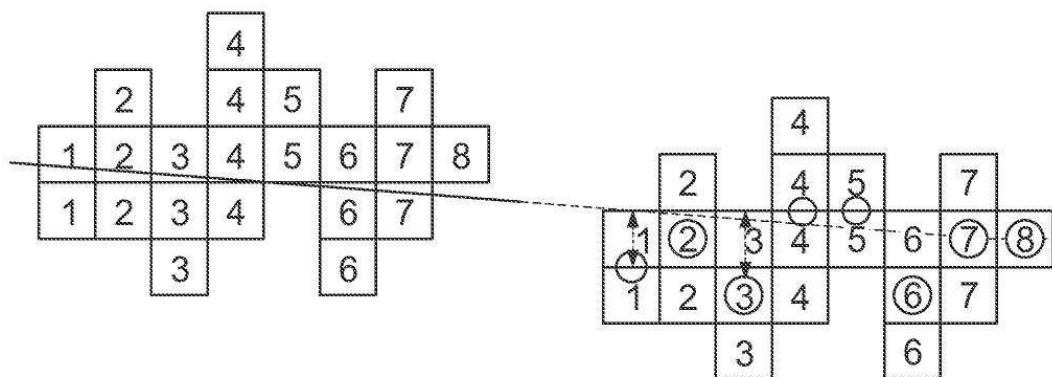
도면9b



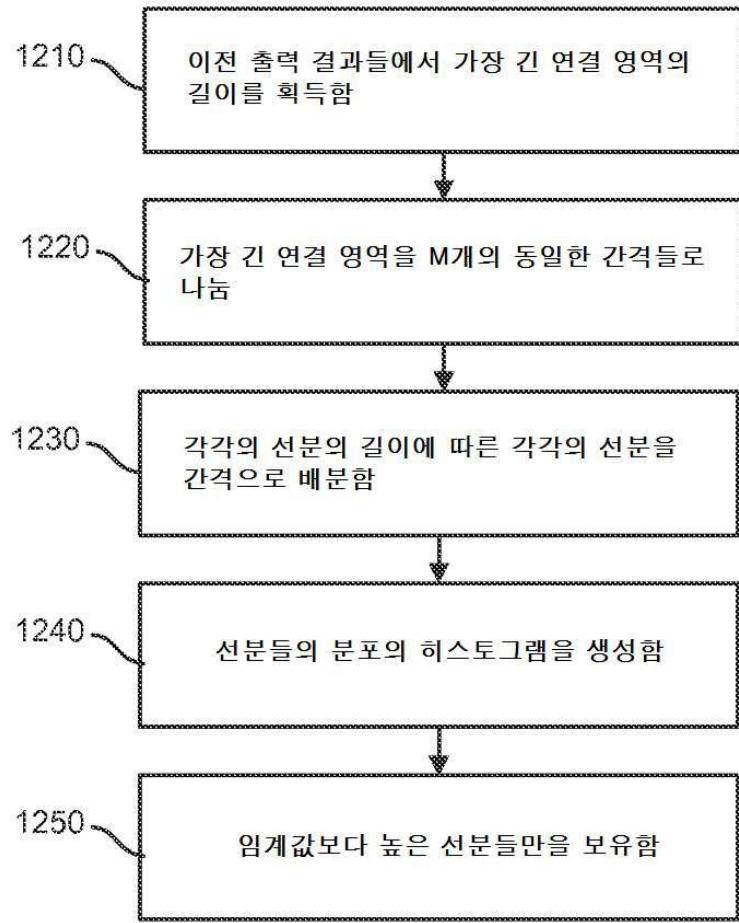
도면10



도면11

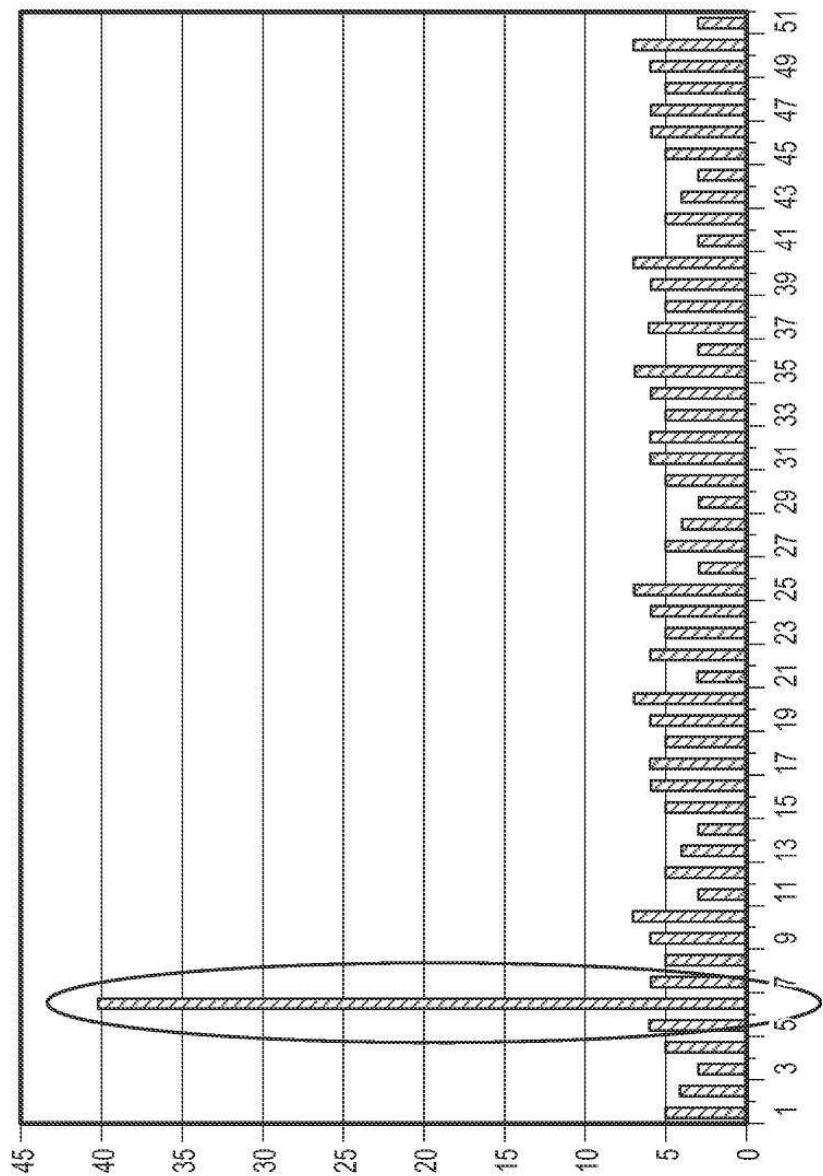


도면12

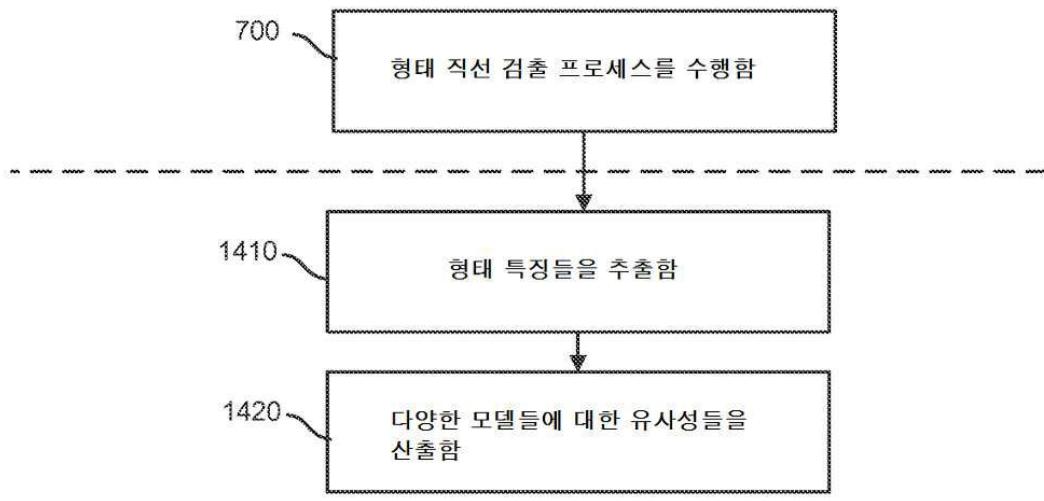


1200

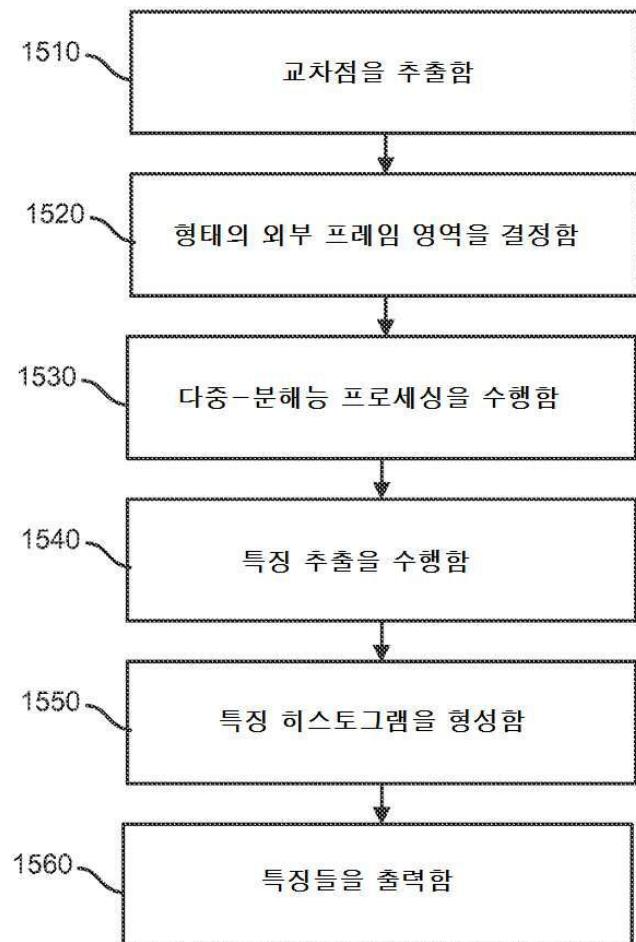
도면13



도면14

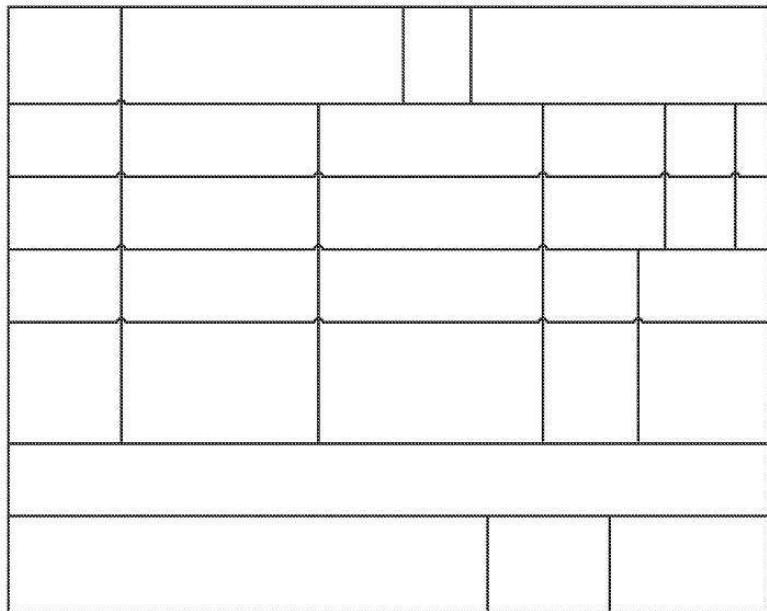
1400

도면15

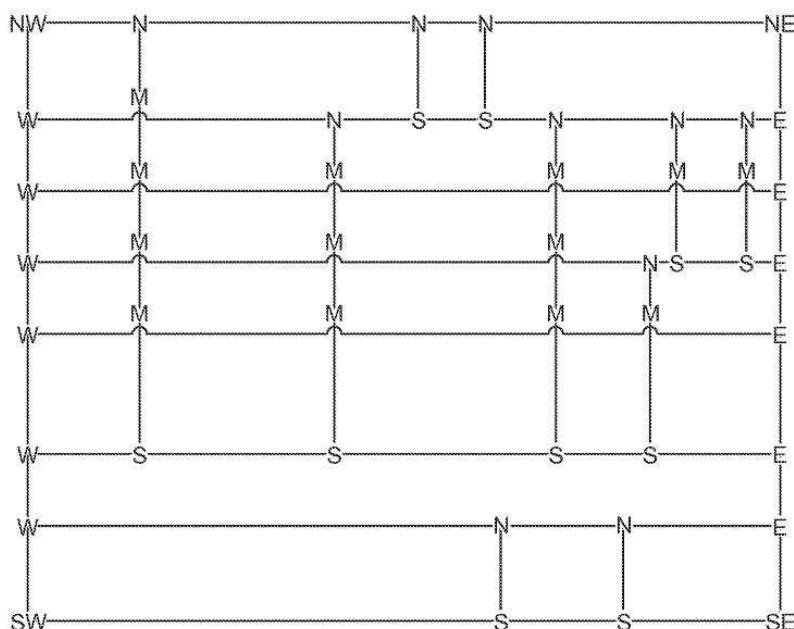
1500

도면16

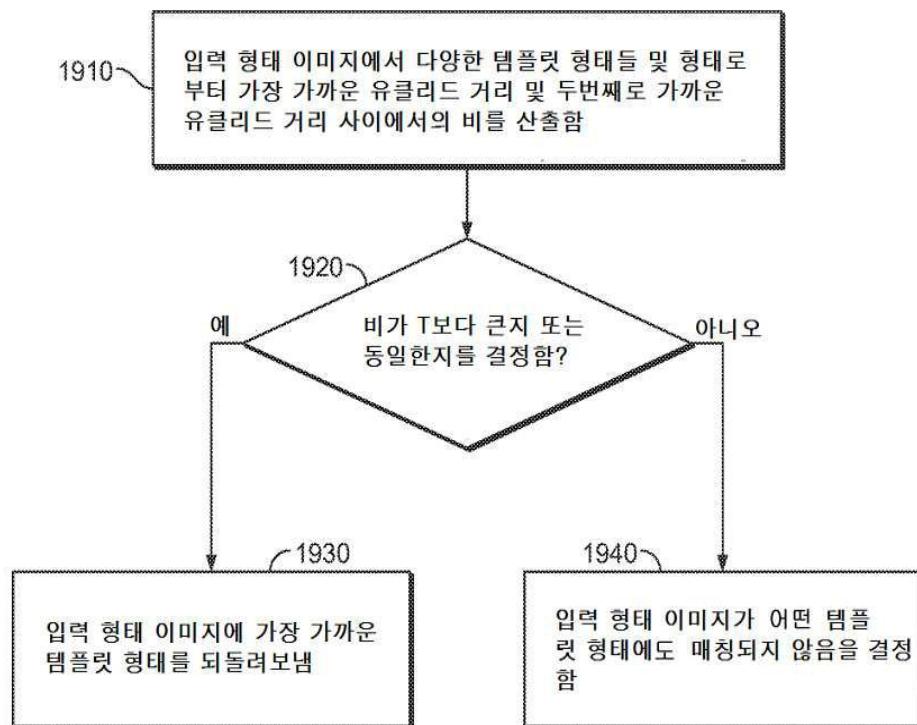
도면17



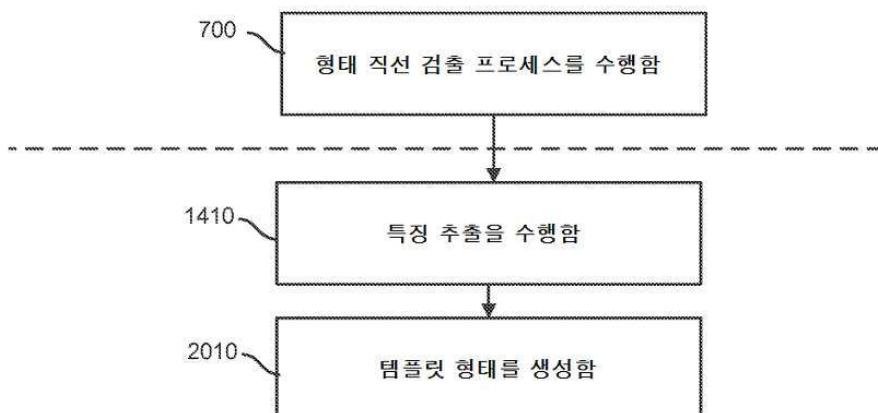
도면18



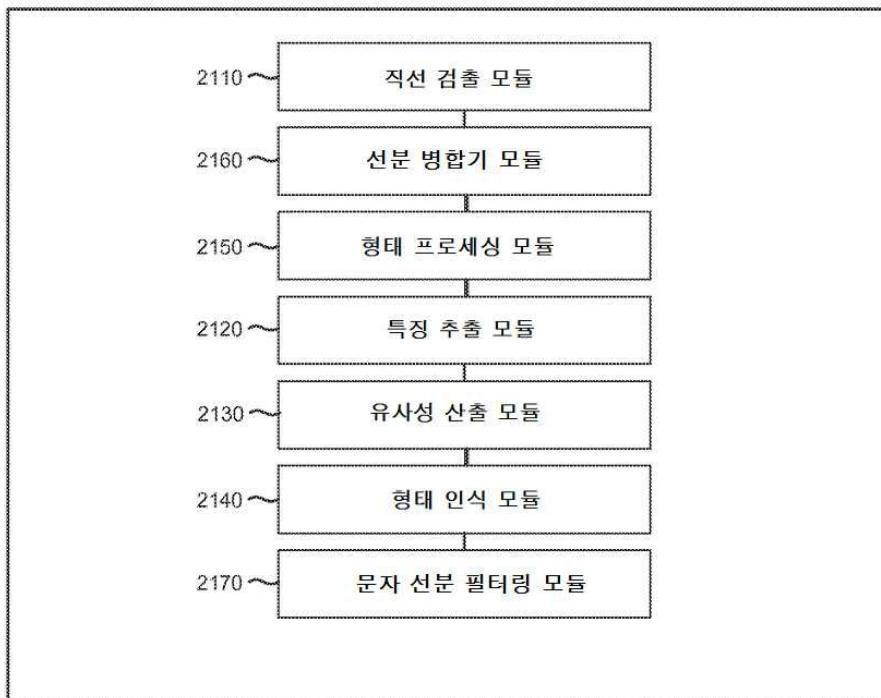
도면19

1900

도면20

2000

도면21



2100

도면22

