

제품 포장라인 검사에 적용 가능한 객체 인식 영상처리 알고리즘 구현

Realization of Image Processing Algorithms for Object Recognition Applicable to Packaging Inspection Processes

김태규*, 이창호**, 안호균***, 윤태성****
Tae-Gyu Kim, Chang-Ho Lee, Ho-Gyun An, Tae-Sung Yoon

Abstract - Using the object recognition processing on the captured images, we can inspect whether a packaging process is performed correctly in real time. So we realized the functions that acquire an image of each state of the packaging process using a camera, extract each object in the image, and inspect the packaging process using the extracted object data. In case an object shape is solid, for object search, a shape-based matching algorithm was used which searches the object utilizing the informations on the shape. In case an object shape is not solid, and is flexible, gray-level difference of the pixels in the limited image region including the object was used to recognize the object.

Key Words : 객체인식, 포장공정, 영상처리

1. 서론

제품 포장라인에서 포장 상황을 영상으로 찍어서 영상을 통하여 객체 인식을 수행하면 객체가 올바르게 포장되고 있는지의 여부를 실시간으로 검사할 수 있다. 본 연구에서는 포장 공정에서 포장 상황을 카메라를 통해 컴퓨터로 읽어 들인 다음 각각의 개체들에 대한 정보를 추출하고 그 정보들을 바탕으로 포장 검사를 수행하기 위한 기능들을 구현하였다. 영상 내에서 특정한 객체를 찾기 위해서는 해당 객체에 대한 정보(객체의 기준 영상 : 템플릿)를 기억해 두었다가 객체 탐색 시 객체 정보와 객체를 탐색하고자 하는 영상을 서로 비교하여 영상 내에서 객체의 정보와 가장 일치하는 영역을 찾는다.[1], [2] 객체의 모양이 일정한 경우에는(즉, 고형의 경우는) 객체의 기준영상과 검사하고자 하는 영상에서 평균 밝기 값을 빼고 정규화 시킨 후 두 영상의 상호 상관계수를 구하여 검사 영상 내에서 해당 객체의 유무를 판단하는 농담정규화 정합(Normalized Gray-level Correlation) 알고리즘을 사용한다. 또한, 객체 영상 내에서 주변의 색상 차이가 큰 부분들을 형태로 추출하고 추출된 형태의 특징들(경계 길이, 곡률, 객체 영역, 객체의 크기 및 무게중심, 모멘트 등)을 객체를 탐색하는 데 이용한다.

객체의 모양이 일정하지 않은 경우(케이블과 같은 유연한 물체) 농담정규화 정합이나 형태 기반 탐색 알고리즘으로는 객체의 탐색이 어렵기 때문에 검사하고자 하는 영상 내에서 객체를 탐색할 영역을 한정시킨 다음에 객체 영상의 히스토그램을 이용하여 적절하게 밝기 임계치를 설정하여 배경 영

역과 객체 영역으로 영상을 분할하고 그 임계치보다 높거나 낮은 수치를 가지는 픽셀들의 수를 기억시킨다. 그런 후 객체 탐색 시 설정된 픽셀 수치와 검사 영역에서의 픽셀 수치를 비교하여(두 픽셀 수치의 차이가 크다면 해당 객체가 존재하지 않는 것으로 간주) 물체의 유무를 판단한다.

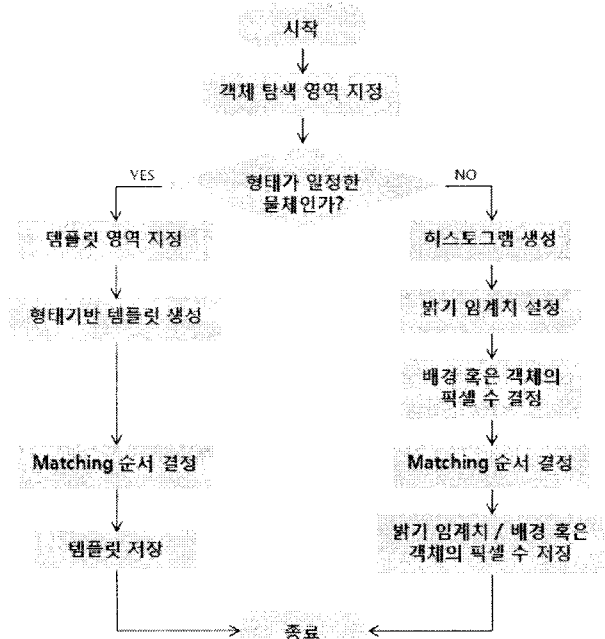


그림 1 객체 정보 생성 순서도

영상 처리 및 객체 인식 알고리즘을 구현하기 위해 C# 언어와 원본 라이브러리[3], [4] 기반에서 GUI 프로그램을 작성

저자 소개

- * 金 泰 奎 : 昌原大學校 電氣工學科 博士課程
- ** 李 昌 鎬 : 昌原大學校 電氣工學科 碩士課程
- *** 安 鎬 均 : 昌原大學校 電氣工學科 工學博士
- **** 尹 泰 星 : 昌原大學校 電氣工學科 工學博士

하였다.

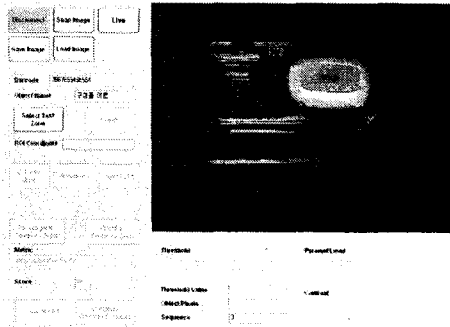


그림 2 원본 라이브러리 기반의 객체인식 인터페이스

2. 형태 기반 객체 탐색

객체 인식을 위해 NGC 알고리즘을 사용하였을 때는 객체의 템플릿과 검사하고자 하는 영상의 밝기가 어느 정도 차이가 나더라도 객체 인식이 가능했지만 상대적으로 수행 속도가 많이 느리고 객체의 템플릿 영상을 생성할 때의 배경과 검사 영상의 배경의 밝기나 컬러가 달라지면 인식 성능이 많이 저하되는 단점이 있었다. 그래서 이번 논문에서는 고품의 객체를 탐색할 때 객체의 형태 정보를 추출하여 객체를 탐색하는 형태 기반 탐색(shape-based matching) 알고리즘을 사용하였다.

영상 내에서 밝기 변화가 큰 두 영역의 경계를 에지(경계선)으로 표현하면 객체의 형태를 표현할 수 있다. 에지는 밝기가 다른 두 영역의 경계이므로 밝기의 기울기의 크기를 계산(1차 미분)하면 영역의 경계를 추출할 수 있다.

영상 내 한 픽셀에서 1차 미분은 영상이 2차원 공간 함수이므로 x, y 두 축에 대한 기울기로 구할 수 있다. 영상 함수 $f(x, y)$ 의 점 (x, y) 에서의 기울기는 다음과 같다.

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (1)$$

x 축과 y 축 방향의 기울기는 각각 다음과 같다.

$$\frac{\partial f}{\partial x} = f(x+1, y) - f(x, y) \quad (2)$$

$$\frac{\partial f}{\partial y} = f(x, y+1) - f(x, y) \quad (3)$$

x, y 두 축 방향의 기울기의 크기는 다음과 같이 계산할 수 있다.

$$\nabla f = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2} \quad (4)$$

식 (4)를 다음 식 (5)와 같이 근사화하면 수식이 보다 간단해지고 구현이 용이하게 된다.

$$\nabla f \approx \left| \frac{\partial f}{\partial x} \right| + \left| \frac{\partial f}{\partial y} \right| \quad (5)$$

기울기의 방향을 나타내는 위상 정보는 다음과 같이 나타낸다.

$$\theta = \tan^{-1} \left(\frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}} \right) \quad (6)$$

인식하고자 하는 객체의 템플릿을 생성하기 위해 먼저 객체를 포함하는 영상을 카메라를 통해 PC로 읽어 들인다. 그 다음 영상 내에서 템플릿으로 생성할 부분을 객체를 포함하도록 하여 지정한 후 템플릿을 생성하기에 앞서 생성될 템플릿을 확인하고 예지로 표현할 최소 밝기 대비를 지정할 수 있도록 에지 추출 알고리즘 구현을 통해 형태 정보를 화면에 표시한다.

형태 기반 객체 탐색을 수행할 때 객체 탐색 속도를 높이기 위해서 영상 피라미드 알고리즘을 구현할 수 있다. 원본 템플릿 영상과 그것을 기반으로 단계별로 축소하여 생성된 템플릿 영상들의 배열을 영상 피라미드라 한다. 원본 영상의 크기를 $2^n \times 2^n$ 이라 하면 축소 레벨 k 에서의 픽셀 $f_k(x, y)$ 는 레벨 $k+1$ 의 영상에서 이웃하는 픽셀 f_{k+1} 로부터 다음과 같이 표현할 수 있다.

$$f_k(x, y) = \frac{1}{4} \sum_{l=0}^1 \sum_{m=0}^1 f_{k+1}(2x+l, 2y+m) \quad (7)$$

여기서 $x, y = 0, \dots, 2^k$ 이다. 형태 정보를 화면에 나타낼 때 원래 크기의 템플릿 뿐만 아니라 식 (7)을 이용해서 크기를 단계별로 축소시킨 템플릿들을 화면에 같이 표시되도록 한다.

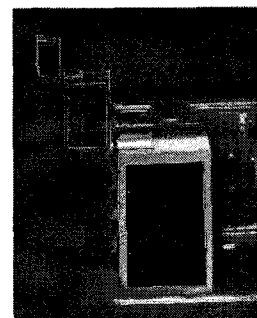


그림 3 영상 피라미드

최소 레벨이 작아질수록 영상의 크기는 작아지고, 이웃한 픽셀들이 하나의 값으로 사상(mapping)되어 고주파 부분이 점차 감쇠되는 특성을 지니게 된다. 이런 식으로 피라미드를 생성하여 템플릿을 만들고 객체를 탐색할 때 제일 낮은 레벨의(제일 크기가 작은) 템플릿을 이용하여 영상의 전 영역에서 탐색을 시도하는데, 크기가 작은 템플릿은 객체의 대략적인 특징만을 가지고 있으며 영상 내에서 객체를 찾으려면 기 위치에서 점차적으로 높은 레벨의 템플릿을 이용하여 객체의 세부 부분을 분석해 나가게 된다. 만약 제일 낮은 레벨의 템플릿을 이용하여 탐색이 실패할 경우 그대로 객체 탐색은 실패하며, 따라서 제일 낮은 레벨의 템플릿이 객체를 잘 표현하도록 객체 영상의 피라미드 레벨을 잘 조절하는 것이 필요하다. 이러한 방식의 장점은 처음에 간략화된 정보를 이용하여 탐색을 시도하며, 탐색 성공 시 그 영역에 대해서만 세부 분석을 수행하므로 탐색 속도를 빠르게 할 수 있다는 것이다.

적당한 최소 밝기 대비와 피라미드 레벨을 정하였다면 템플릿을 파일로 저장한다. 객체 탐색을 할 때 저장된 템플릿 파일을 불러들인 다음 아래 식과 같이 표현되는 MSE(Mean Square Error)를 이용해서 탐색 영상의 전 영역에서 객체 탐색을 수행할 수 있다.

$$MSE = \frac{1}{MN} \sum_{i=0}^M \sum_{j=0}^N (f(x_i, y_j) - T(x_i, y_j))^2 \quad (8)$$

식 (8)에서 M과 N은 템플릿 영상의 각각 가로와 세로 길이를 나타내며, f는 탐색 영상, T는 템플릿, x와 y는 영상을 구성하는 픽셀의 좌표를 나타낸다. MSE는 탐색 영상과 템플릿의 차이의 제곱의 평균을 나타내는데, 템플릿과 영상이 일치할수록 MSE가 작아지며, 적당한 임계치를 미리 설정한 다음 임계치에 따라서 일치 여부를 판단한다.

3. 히스토그램을 이용한 객체 탐색

히스토그램을 이용하여 객체를 탐색하기에 앞서 먼저 객체 탐색을 위해 필요한 정보들을 수집한다. 먼저 영상을 획득하여 화면에 표시한 다음 객체 탐색 영역을 지정한다. 영역을 지정하고 그 영역 정보를 저장하는 방법은 여러 가지가 있으나 여기서는 영역을 사각형 모양으로 지정한 후 사각형의 좌측 상단과 우측 하단 꼭지점의 좌표를 입력받아 저장하였다.

검색 영역을 지정하면 그 영역에서의 영상의 히스토그램을 생성하는데, 그 전에 먼저 영상을 흑백으로 전환하는 과정이 필요하다. 영상을 흑백으로 전환한 후 영상의 히스토그램을 만든다. 히스토그램은 길이가 256인 배열 형태가 되는데, 배열의 각 요소의 순서는 0에서 255까지의 픽셀의 그레이 스케일(gray scale) 값을 뜻하며 각 요소에는 그 요소의 개수, 즉 그레이값을 가지는 픽셀들의 수가 저장된다. 그 후에 밝기에 따라 객체 영역과 배경 영역을 분할하도록 밝기 임계치를 설정하여 임계 밝기보다 밝거나 혹은 어두운 픽셀들을 선택하여 화면에 나타내도록 하였다(본 연구에서는 기준보다 밝은 픽셀들을 선택). 이렇게 하면 화면을 보면서 기준치를 조절하여 영상에서 객체 영역과 배경 영역이 제대로 분할되고 있는지 확인할 수 있다.

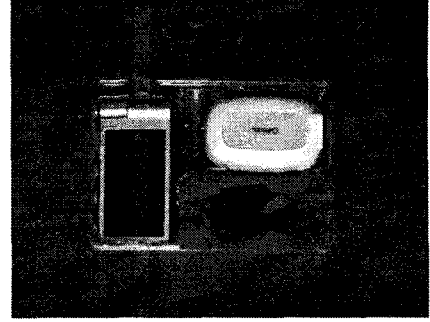


그림 4 객체 영역과 배경 영역 분할을 위한 밝기 기준치 조절

적절히 기준치를 조절하였으면 기준치와 그 아래, 혹은 그 위의 밝기값을 가지는 픽셀들의 합을 구하여 저장해 둔다(히스토그램 배열에서 기준치 순서보다 앞, 혹은 뒤의 순서의 요소들의 수치(픽셀 수)들을 모두 더하면 된다).

4. 결론

본 연구에서는 제품의 포장 라인에서 얻어진 영상 내에서 객체의 형태 정보를 또는 히스토그램을 이용한 영상의 각 픽셀들의 밝기 분포의 차이를 이용하여 객체를 탐색하고 인식하는 영상처리 알고리즘을 구현하였다. 객체의 모양이 일정한 경우에는 객체의 형태 정보를 이용해서 형태 기반 탐색 알고리즘을 구현하는 것이 가능했지만 아이폰이나 케이블같이 모양이 일정하지 않고 유연한 객체들은 특정한 형태 정보를 이용한 탐색 알고리즘을 구현할 수 없게 된다. 따라서 영상 내에서 객체를 이루는 픽셀들과 배경 픽셀들 사이의 밝기의 차이를 이용해서 객체의 유무를 판단하였다. 그러나 이 방법은 일정한 형태 정보를 이용한 탐색 알고리즘보다 정확성이 떨어지기 때문에 제한적으로만 구현해야 할 것이며, 단순 밝기의 차이를 이용하는 것보다 객체의 컬러 정보를 이용한다면 보다 정확한 탐색 알고리즘 구현이 가능하리라 본다.

참 고 문 헌

- [1] 이홍규, "디지털 영상처리 이론 및 구현 - Visual C++ 접근 방법", 사이텍미디어, 2007. 9.
- [2] E. R. Davies, "Machine Vision", Morgan Kaufmann, 2005.
- [3] "HALCON Solution Guide I - Basics", MVTec Software GmbH, April 2008.
- [4] "HALCON Solution Guide II - Advanced Topics", MVTec Software GmbH, April 2008.

"본 논문은 2008년 산학 공동기술개발지원사업의 연구결과입니다."