

부품 이미지 인식을 위한 HTM 네트워크 훈련 시스템 개발

이대한[†], 배선갑^{**}, 서대호^{***}, 강현석^{****}, 배종민^{*****}

요 약

여러 품종을 소량으로 생산하는 소형 공장에서 불량품으로 인한 손실을 줄이기 위하여 부품의 양불량을 판단하는 시스템의 개발이 필요하다. 그러한 시스템은 계층형 시간적 메모리(HTM : Hierarchical Temporal Memory) 기술을 이용하여 개발할 수 있다. HTM은 인간 두뇌의 신피질(neocortex)의 동작 원리를 기계학습에 접목시킨 모델이다. HTM 기반의 기계학습 시스템을 사용하기 위해서는 훈련된 HTM 네트워크를 개발해야 하는데, 이를 위해서는 HTM 이론에 대한 지식이 필요하다. 본 연구는 이 HTM 기술을 부품의 이미지 인식에 적용하여 부품에 대한 양·불량을 판별하는 시스템에서, HTM 네트워크 개발을 지원하는 훈련시스템의 설계와 구현을 제시한다. 이 시스템은 HTM 이론에 대한 지식이 없어도 작업현장의 기술자가 HTM 네트워크를 정확히 훈련시킬 수 있으며, 부품에 대한 모든 종류의 HTM 기반의 판정시스템에 그대로 적용될 수 있다.

Development of an HTM Network Training System for Recognition of Molding Parts

Dae-Han Lee[†], Sun-Gap Bae^{**}, Dae-Ho Seo^{***},
Hyun-Syug Kang^{****}, Jong-Min Bae^{*****}

ABSTRACT

It is necessary to develop a system to judge inferiority of goods to minimize the loss at small factories in which produces various kinds of goods with small amounts. That system can be developed based on HTM theory. HTM is a model to apply the operation principles of the neocortex in human brain to the machine learning. We have to build the trained HTM network to use the HTM-based machine learning system. It requires the knowledge for the HTM theory. This paper presents the design and implementation of the training system to support the development of HTM networks which recognize the molding parts to judge its badness. This training system allows field technicians to train the HTM network with high accuracy without the knowledge of the HTM theory. It also can be applied to any kind of the HTM-based judging systems for molding parts.

Key words: HTM, Machine Learning(기계학습), Image Recognition(이미지인식), Training system(훈련 시스템)

※ 교신저자(Corresponding Author): 배종민, 주소: 경남 진주시 가좌동 900 경상대학교 자연과학대학 컴퓨터과학부(660-701), 전화: 055-751-5995, FAX: 055-762-1944, E-mail: jmbae@gnu.ac.kr

접수일: 2010년 4월 8일, 수정일: 2010년 7월 20일

완료일: 2010년 9월 6일

[†] 정회원, 경상대학교병원

(E-mail: neokshacker@naver.com)

^{**} 준회원, 경상대학교 대학원 컴퓨터과학과
(E-mail: bsgap@hanmail.net)

^{***} 정회원, 경상대학교 대학원 컴퓨터과학과
(E-mail: chorokh@naver.com)

^{****} 종신회원, 경상대학교 컴퓨터과학부 교수
(E-mail: hskang@gnu.ac.kr)

^{*****} 정회원, 경상대학교 컴퓨터과학부 교수

1. 서 론

프레스 기계를 이용하여 여러 종류의 부품들을 소량 생산하는 중소 규모의 공장에서는 생산된 부품의 불량여부 검사를 사람의 눈에 의존하는데 대부분 전수검사를 수행한다. 그런데 한 사람이 여러 시간 동안 많은 양을 검사하기 때문에 집중력이 부족해져 불량여부를 잘못 판단할 수 있어서 검사 결과에 대한 신뢰성을 저하시킨다. 또한 생산이 이루어진 이후에 검사 작업이 이루어지므로 이미 발생한 많은 양의 불량품에 대한 시간적, 금전적으로 손실을 입게 된다.

이를 해결하기 위해서는 적은 비용으로 효율적인 양품과 불량품을 판별하는 시스템이 필요하다. 이는 첫째, 생산 제품의 이미지를 현장에서 수집하고 학습한 후 실시간으로 생산된 제품의 이미지를 인식하여 판별하고, 둘째, 다양한 가변 환경에서도 사람이 인식할 때와 같은 수준의 인식률이 요구되며, 셋째, 생산하는 제품이 바뀔 때마다 쉽게 적용시킬 수 있어야 한다. 이러한 요건을 만족시키기 위해서는 인간의 학습 능력과 이미지 인식 능력에 비견할 수 있는 새로운 기술이 필요하다. 기존의 이미지 프로세싱을 통한 부품검사시스템은 특정 부품 하나를 목표로 개발된다[1]. 이 경우 여러 품종을 소량으로 개발하는 중소 제조업체에서 각 부품에 맞는 검사시스템을 도입하는 것은 쉬운 일이 아니다.

인간의 인식 능력은 주로 인간 두뇌의 외부를 둘러싸고 있는 얇은 막인 신피질의 기능에서 비롯된다. 그동안 신피질의 동작 원리를 흉내 내는 시스템을 개발하려는 노력이 많이 이루어져 왔다[2]. 그 중 신피질의 동작 기제에 좀 더 가까운 것이 최근 Jeff Hopkins가 제안한 계층적 시간적 메모리(Hierarchical Temporal Memory(HTM)) 이론이다[3]. HTM 이론은 노드들을 계층 네트워크(Hierarchy network)로 구성한 다음 실세계의 시공간적 패턴 정보를 이용해 학습을 한 후, 효율적으로 지능적 판단을 할 수 있는 방법론이다. HTM 이론은 사물 인식, 비디오 인식, 주식시장 예측 등 다양한 인식 분야에 활발하게 적용되고 있다[4]. 이에 따라 HTM 이론을 이용하여 생산부품의 불량 여부를 판정하는 시스템도 실제로 개발되었다[5].

하지만 이러한 판정 시스템이 효과적으로 현장에 적용되기 위해서는 부품 이미지 인식 훈련 과정을

효과적으로 수행할 수 있는 훈련 시스템의 개발이 필요하다. 이때 필요한 기능은 다음과 같다. 첫째, HTM 이론에 대한 지식을 갖춘 전문가가 아닐 경우에 특정 응용에 적합한 HTM 네트워크구조를 직접 만드는 것은 어렵기 때문에 개발할 훈련 시스템은 HTM이 부품 이미지 인식에 최적화된 HTM 네트워크 구조를 제공하거나 제안할 수 있어야 한다. 둘째, HTM 훈련 과정에서 생성된 HTM 네트워크의 관리나 훈련된 HTM 네트워크의 정확도를 모두 사용자가 직접 관리할 수 있도록 해야 한다. 셋째, HTM의 훈련 과정에서 필요시 원하는 훈련 단계로 돌아가고 시점부터 HTM 네트워크 훈련시킬 수 있게 해주는 피드백 기능이 필요하다.

이에 따라서, 본 논문에서는 HTM 이론을 구현한 플랫폼인 NuPIC을 활용하여 제품 이미지를 이용한 양물 판정 시스템을 개발하고 운영함에 있어 필요한 훈련된 HTM 네트워크의 개발 과정의 효율성을 증대시킬 수 있는 HTM 네트워크 훈련 시스템을 제시한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 HTM 이론을 소개하고 이를 이용한 시스템 개발 과정을 설명한다. 3장에서는 HTM 기반의 부품 이미지 인식 응용 분야에 적합한 훈련 시스템을 설계하고 4장에서는 시나리오에 따라 HTM 네트워크를 훈련시키는 과정을 통해 본 훈련 시스템의 구현 결과를 제시한다. 5장에서는 HTM 기반의 기존 훈련도구들과 비교하여 본 시스템의 평가를 제시한다. 마지막으로 6장에서는 결론과 향후과제를 제시한다.

2. 관련 연구

HTM 네트워크를 훈련시키는 기존 도구로는 RunExperiment[6]와 Vitamin D사의 Vitamin d toolkit[7]이 있다.

RunExperiment는 이미지 인식 분야를 대상으로 한 HTM 네트워크 훈련 도구이다. RunExperiment를 수행시키면 먼저 param.py로부터 정의된 HTM 네트워크 구조를 읽어 초기 HTM 네트워크를 생성시킨 후 지정된 학습 데이터 집합을 읽어 HTM 네트워크를 훈련시킨다. 훈련 과정과 테스트 과정은 RunExperiment를 실행할 때 텍스트 모드 또는 GUI 모드를 지정할 수 있다. 훈련이 종료되면 훈련된 HTM 네트워크가 생성되고 이어서 params.py에 지

정된 테스트 데이터 집합을 사용하여 훈련된 HTM 네트워크의 정확도를 측정한다. 테스트가 종료되면 결과는 result.txt 파일에 저장되고 프로그램이 종료된다.

이러한 RunExperiment는 그래픽 사용자 인터페이스 지원이 부족하여 HTM 네트워크를 훈련시키기 어렵다. 특히 훈련과 테스트하는데 필요한 정보를 파이썬 스크립트로 작성해야 하기 때문에 파이썬 스크립트 문서를 텍스트 문서 편집기를 사용하여 직접 수정하면서 훈련과 테스트를 진행해야 한다.

Vitamin d toolkit은 Vitamin D사에서 제공하는 HTM 네트워크 훈련 도구이다. Vitamin d toolkit은 HTM 네트워크 개발 시 다양한 기능을 제공한다. 특히 사용자가 대부분의 기능을 그래픽 사용자 인터페이스를 통해 접근할 수 있고 HTM 네트워크 훈련 결과를 시각적으로 볼 수 있기 때문에 NuPIC에서 제공하는 훈련 도구인 RunExperiment 보다 사용하기 편리하다.

Vitamin d toolkit을 이용하여 HTM 네트워크를 훈련시키기 위해서는 훈련용 데이터 집합을 선택해야 한다. 먼저 훈련용 데이터 집합을 특정 디렉터리에 넣어 둔다. 다음 'File' 메뉴에서 'Train Network'를 선택하거나 툴바에서 'Train' 버튼을 눌러 훈련용 데이터 집합의 위치를 선택하면 HTM 네트워크 훈련 과정이 진행된다.

훈련이 끝난 후 훈련된 HTM 네트워크를 분석해 보고자 할 때는 툴바의 'Training' 메뉴를 선택하면 HTM 네트워크가 훈련된 상황을 히스토그램, 동시 발생 수, 그룹 형성 상태, TAM, 속성 방식으로 볼 수 있다. HTM 네트워크를 훈련시킨 후 훈련 결과가 만족스럽지 않을 경우 그 원인을 분석하여 디버깅을 수행해야 한다. Vitamin d toolkit은 디버깅시 입력 데이터에 대해 HTM 네트워크가 레벨별로 추론한 값을 다양한 방법으로 자세히 보여주어 사용자가 좋은 훈련 전략을 만들 수 있도록 도와준다.

위 두 가지 훈련 도구는 HTM을 훈련하는데 있어 일반적으로 사용할 수 있는 범용 훈련 도구이다. 이 도구들은 응용의 종류에 관계없이 대부분의 HTM 네트워크를 훈련시키는데 사용될 수 있지만 생산 현장 실무자가 사용할 경우 몇 가지 단점이 있다.

첫째, 특정 응용 분야를 목표로 HTM 네트워크를 훈련시키는 경우 해당 분야에 적합한 HTM 네트워크 구조가 있을 수 있지만 기본 구조를 제안해 주지

않는다. 게다가 HTM 이론을 바탕으로 하는 HTM 네트워크 구조 생성은 전문기술을 필요로 하는데 현장 기술자들이 이를 활용하는 것을 기대하기는 어렵다.

둘째, Vitamin d toolkit은 HTM 네트워크의 훈련과 디버깅에 대해 뛰어난 기능을 제공하지만 HTM 전문가가 아닌 현장 기술자의 경우 오히려 훈련 도구를 사용하기 어렵게 만드는 요소가 된다.

셋째, RunExperiment는 훈련에 사용되는 주요 내용들을 사용자가 텍스트 파일을 직접 수정해야 하는 등 사용자 인터페이스가 불편하다.

본 논문에서는 이러한 문제점들을 해결한 훈련 시스템을 제시한다.

3. HTM과 HTM 네트워크

먼저 개발에 사용된 HTM 기술과 HTM 네트워크의 전반적인 내용을 기술한다.

3.1 HTM 이론

HTM 이론은 기존의 경험적 탐색을 기본으로 하는 인공지능(AI)이나 단순 뉴런들의 연결로 보는 인공신경망(ANN)과는 달리, 시스템을 일정한 공통 연산을 수행하는 메모리 노드들의 계층 네트워크(hierarchy network)로 구성한다[2,3]. 이때, 네트워크의 기본 단위인 노드(node)는 신피질(neocortex)의 기본 단위인 신피질 칼럼(neocortex column)에 해당하는 것으로 본다. 이렇게 구성된 HTM 네트워크는 실세계의 시간적 패턴 정보와 공간적 패턴 정보를 이용하여 효율적으로 지능적 판단을 할 수 있게 한다. 또한, HTM은 일종의 메모리 시스템으로 문제마다 각자 다른 알고리즘을 수행하는 것이 아니라 문제를 해결하는 방법을 학습한다. 그리고 학습 과정에서 감각 신호의 특성들 사이의 시공간적 관계성이 계층적 메모리 구조에 형성된다. 이러한 학습이 끝나면 새로운 입력 패턴을 인식할 수 있게 된다.

3.2 HTM 네트워크

HTM에서 네트워크는 노드들의 계층으로 구성된다. 데이터는 계층의 바닥으로부터 감각(Data Sensor)을 통해 입력되고, 입력된 데이터는 노드들의 계층 구조를 통해 위로 올라가면서 처리된다. 이때 계

층의 각 노드는 학습과 추론의 두 단계로 이루어진 알고리즘을 사용한다[4,8].

학습 단계에서는 네트워크가 훈련 패턴에 노출되어 범주(Category)들에 패턴들을 사상시키는 모델을 구축하고, 추론 단계에서는 네트워크가 이전에 본 일이 없는 새로운 패턴에 대해 범주들에 대한 간단한 분포를 생성한다. 전형적인 노드들의 계층적 구조를 그림 1의 (가)에 개략적으로 나타내었다. 그림에서 보는 것과 같이 HTM 네트워크는 몇 가지 유형의 노드들로 구성된다[3]. 이들 중 가장 핵심적인 것이 Zeta1 노드와 Zeta1Top 노드이다. 그 중에서 Zeta1 노드가 계층의 대부분을 구성하게 된다. Zeta1 노드는 그림 1의 (나)와 같이 정보를 공간적 풀러(Spatial pooler)와 시간적 풀러(Temporal pooler)의 두 가지 구성요소로 처리하게 된다. 공간적 풀러는 데이터의 공간적 패턴을 분류하는 역할을 수행한다. 즉, 입력 벡터들의 분산된 표현들을 분류하는 방법으로 학습한다. 시간적 풀러는 데이터의 동시발생(Coincidence)을 기초로 시간적 패턴을 그룹화 하는 알고리즘을 수행한다. 이는 추론 결과가 이전 입력 값뿐만 아니라 최근의 입력 값을 기초로 하여 계산하는 시간 기반의 알고리즘이다. 그리고 그림 1의 (다)에 나타낸 Zeta1TopNode는 공간적 풀러와 통제 사상기로 이루어져 있다. Zeta1TopNode의 공간적 풀러는 Zeta1Node와 같은 역할을 하며, 통제 사상기는 범주 정보를 이용하여 최종적으로 판단한다.

3.3 HTM 네트워크의 개발 과정

일반적인 HTM 네트워크의 개발 과정은 다음과 같은 순서로 이루어진다[10].

첫째, 주어진 문제를 분석한다. 문제 분석 단계에서는 해결하고자 하는 문제를 정확하게 정의한다. 문제 분석시 인식하고자 하는 대상에 대한 유형(카테고리)을 구분한다. 둘째, HTM 네트워크 구조를 설계한다. 문제 분석 결과를 바탕으로 문제를 해결하기에 가장 적합하다고 판단되는 HTM 네트워크의 계층 수, 각 계층별 노드 수, 계층 간의 연결 방법을 결정한다. 셋째, HTM 네트워크를 훈련시킬 데이터와 훈련된 HTM 네트워크의 정확도를 테스트할 테스트 데이터를 구분하여 수집한다. 넷째, 훈련시에 입력되는 데이터들을 HTM 노드가 분석하는데 사용되는 파라미터 값을 입력한다. 다섯째, 훈련되지 않은 HTM 네트워크를 생성한다. 여섯째, 훈련용 데이터를 사용하여 HTM 네트워크를 훈련시킨다. 일곱째, 훈련된 HTM 네트워크를 테스트용 데이터를 사용하여 그 정확도를 테스트한다. 그런데 테스트 결과가 만족스럽지 않은 경우 다시 앞의 작업을 반복한다. 그리고 반복 과정 중 최적의 결과를 얻으면 하나의 HTM 네트워크 개발이 완료된다.

3.4 인식 정확도에 영향을 미치는 요소

HTM 네트워크가 패턴인식을 하는데 있어서 인식에 가장 큰 영향을 미치는 파라미터는 Max-Distance, Sigma가 있다.

MaxDistance는 하나의 기준 센서로부터 원본 이미지와 비교 대상인 입력 이미지의 벡터 사이의 유클리디안(Euclidean) 거리의 최대값이다. 이는 학습하는 동안에 입력된 벡터들이 한 기준점에서 Max-Distance 값을 설정하면 입력되는 이미지 센서의 벡터들의 거리를 각각 계산하여 이것과 비교한다. 계산

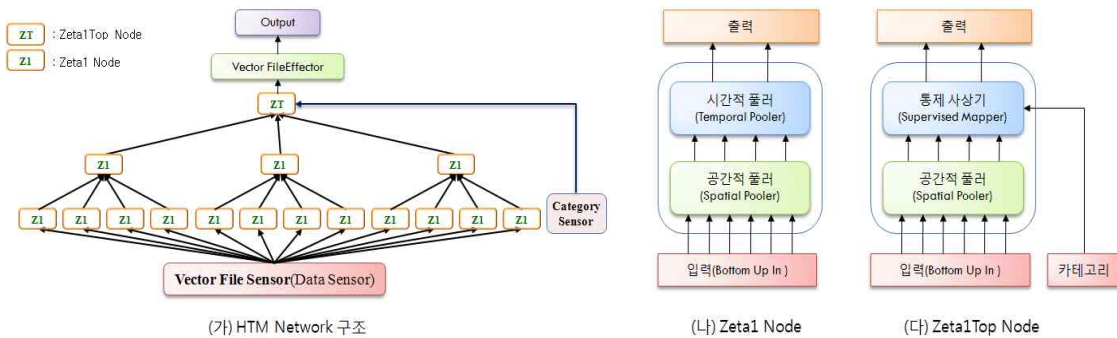


그림 1. HTM 네트워크의 구조

된 값이 MaxDistance의 값보다 작을 경우는 같은 동시발생으로 취급하고, MaxDistance 보다 클 경우는 새로운 동시발생으로 취급한다[8]. 따라서 Max-Distance를 크게 설정해 줄 경우, 속도는 빠르지만 정확한 비교는 어렵다. 반면 MaxDistance를 작게 설정할수록 학습 속도는 느리지만 좀 더 정확한 비교를 할 수 있다. 뚜렷이 구분이 되는 이미지를 인식하는 응용에서는 비교적 MaxDistance 값이 커야 높은 정확도를 나타낸다. 반면에 미세한 차이를 보이는 이미지를 구별해 내는 응용에서는 MaxDistance 값이 작아야 한다.

Sigma는 추론의 과정에서 필요한 파라미터로서 동시발생 사건의 정규 분포의 표준편차이다. 이는 동시에 발생하는 경우의 수의 범위를 지정한다. 이 값이 작은 것은 동시발생 사건이 밀집되어 있음을 의미하며, 큰 값은 넓게 분포함을 뜻한다. 대개 추론시 잡음이 많은 이미지일 경우는 이 값을 크게 설정하고, 깨끗한 이미지일 경우는 이 값을 작게 설정하면 정확도가 높게 나타난다.

4. 부품 이미지 인식을 위한 훈련 시스템의 설계

4.1 시스템 구조

개발된 훈련 시스템의 개괄적인 구조는 그림 2와

같다. 사용자는 사용자 인터페이스를 통해 훈련 프로그램을 제어하며, NuPIC은 훈련 프로그램에 의해 호출되어 HTM 네트워크를 생성하고 훈련 프로그램이 전달한 데이터를 이용하여 HTM 네트워크를 훈련시키거나 추론을 수행한다.

4.2 모듈별 기능

4.2.1 훈련용 데이터 집합과 테스트용 데이터 집합

그림 2에서 ‘훈련용 데이터 집합’과 ‘테스트용 데이터 집합’은 HTM 네트워크를 훈련시키거나 훈련된 HTM 네트워크의 정확도를 측정하는데 사용된다. HTM 네트워크를 훈련시키기 전에 사용자는 ‘학습용 데이터 집합’과 ‘테스트용 데이터 집합’을 수집해야 한다. ‘학습용 데이터 집합’은 NuPIC을 통하여 HTM 네트워크 훈련을 위해서 사용될 데이터들의 집합이다. ‘테스트용 데이터 집합’은 훈련되어진 HTM 네트워크의 정확도 테스트를 위한 데이터의 집합이다. 학습용 데이터 집합과 훈련용 데이터 집합을 수집할 때는 제품 이미지의 불량 여부와 불량 유형을 구분하여 수집해 두어야 하며 학습용 데이터 집합과 테스트용 데이터 집합의 유형 구분은 같아야 한다.

4.2.2 카테고리 프리셋 제어기

카테고리 프리셋 제어 모듈은 카테고리 프리셋을

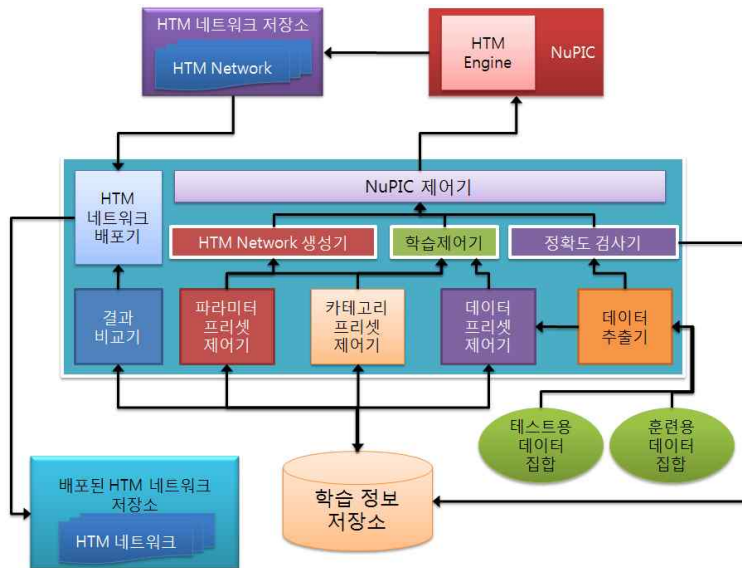


그림 2. 훈련 시스템의 구조

생성하고 삭제하고 저장하는 모듈이다. 카테고리 프리셋이란, 제품을 학습하기 위해 사용될 양품 유형과 불량품 유형들 중 일부를 묶어둔 집합을 말한다. 여기에서 일부라는 것은 하나의 제품에 알맞은 HTM 네트워크를 생성하기 위해 카테고리 프리셋을 정의할 때, 양불 유형 중 정확도 테스트 결과의 일부 유형이 이미지를 통해 양불 여부를 인식할 수 없을 정도의 작은 차이만 있어 HTM 네트워크의 전체적인 정확도를 감소시킬 경우에 해당 유형을 제외시키는 등 사용자가 의도적으로 하나 이상의 양불 유형을 가감할 수 있다는 것을 말한다. 따라서 카테고리 프리셋은 경우에 따라 하나의 제품에도 여러 개가 생성될 수 있다.

4.2.3 데이터 프리셋 제어기

데이터 프리셋 제어기는 트레이닝 데이터 프리셋을 생성하고 삭제하고 저장하는 모듈이다. 트레이닝 데이터 프리셋이란, 카테고리 프리셋에 설정된 제품의 양불 유형들에 대해 각각 ‘학습용 데이터 집합’ 중 어느 것과 대응되는지를 지정하고 ‘학습용 데이터 집합’에서 추출될 실제 데이터 수를 명시하거나 유형별로 그 비율을 지정할 수 있다. 생성된 트레이닝 데이터 프리셋은 하나의 카테고리 프리셋에 종속되며, 반대로는 하나의 카테고리 프리셋으로 훈련 데이터를 달리하며 훈련시킬 수 있어야 하기 때문에 하나의 카테고리 프리셋은 여러 트레이닝 데이터 프리셋을 가질 수 있다.

4.2.4 데이터 추출기

데이터 추출기는 두 가지 목적으로 사용한다. 첫 번째는 훈련시에 사용할 데이터를 추출하는 것이다. 훈련용 데이터를 추출하기 위해서는 데이터 프리셋에 설정된 정보를 읽어 ‘학습용 데이터 집합’에서 해당 카테고리의 훈련용 데이터가 저장된 경로를 찾는다. 다음으로 해당 경로에서 지정된 카테고리별 비율이나 데이터 수만큼 추출한다. 두 번째는 테스트 시에 사용할 데이터를 추출하는 것이다. 테스트 데이터 추출은 먼저 카테고리 프리셋에 설정된 정보를 읽어 ‘테스트용 데이터 집합’에서 해당 카테고리의 테스트 데이터가 저장된 경로를 찾는다. 다음으로 정확도 검사기가 보낸 정보를 읽어 해당 경로로부터 카테고리별 비율 또는 데이터 수에 맞는 데이터를 추출한다.

4.2.5 파라미터 프리셋 제어기

파라미터 프리셋 제어기는 파라미터 프리셋을 생성하고 저장하고 삭제하는 모듈이다. HTM 네트워크를 처음 생성할 때 훈련에 관련된 파라미터를 설정해야 하는데 HTM이 주어진 입력 데이터를 어떻게 학습해야 하는지 지정한다. HTM 네트워크가 NuPIC을 통해 입력된 데이터를 지정된 훈련 파라미터 값에 의해 학습하고 훈련된 HTM 네트워크를 생성한다. 이때 주어진 훈련 파라미터가 실제 제품 이미지로 양불 여부를 판단할 때 판단 결과의 정확도를 결정하게 된다. 즉, 어떻게 훈련을 시키느냐에 따라서 정확도가 달라진다. 주요 훈련 파라미터로는 Max Distance, Sigma, Iteration이 있으며, 세 가지 훈련 파라미터들의 값을 하나로 묶어 관리하는데 이를 파라미터 프리셋이라 한다. 파라미터 프리셋은 다른 프리셋에 종속적이지 않아 별도로 관리된다.

4.2.6 HTM 네트워크 생성기

HTM 네트워크 생성기는 훈련되지 않은 초기 HTM 네트워크를 생성한다. 초기 HTM 네트워크를 생성하기 위해서는 세 가지 정보가 필요한데, HTM 네트워크 구조, 학습할 카테고리 수, 훈련 파라미터 값이 필요하다. 첫 번째 정보인 HTM 네트워크 구조는 사용자가 직접 설정하기 어렵기 때문에 본 응용에서는 여러 테스트를 거쳐 부품 이미지 인식에 가장 적합하다고 판단된 HTM 네트워크 구조를 제공한다. 두 번째 정보인 학습할 카테고리 수는 사용자가 선택한 카테고리 프리셋으로부터 읽는다. 세 번째 정보인 훈련 파라미터 값은 사용자가 선택한 파라미터 프리셋으로부터 읽는다. 이 세 가지 정보가 수집되면 HTM 네트워크 생성 모듈은 NuPIC에 HTM 네트워크 생성 명령을 내린다.

4.2.7 학습 제어기

학습 제어기는 HTM 네트워크를 훈련시킨다. 학습 제어기가 HTM 네트워크를 훈련시키는 과정은 다음과 같다. 먼저, 학습 제어기는 사용자가 선택한 카테고리 프리셋, 데이터 프리셋 정보를 바탕으로 HTM 네트워크 생성기가 만든 훈련되지 않은 HTM 네트워크를 선택한다. 다음으로, NuPIC을 호출하면서 HTM 엔진이 지정된 HTM 네트워크를 훈련시킬 수 있도록 불러온다. 마지막으로, 데이터 추출 모듈

에 의해 추출된 훈련 데이터 집합을 NuPIC에 전달하여 HTM 엔진이 HTM 네트워크를 훈련시키게 한다. 훈련이 끝나면 훈련된 HTM 네트워크가 생성되고 학습 제어기는 훈련된 HTM 네트워크를 HTM 네트워크 저장소에 저장한다.

4.2.8 정확도 검사기

정확도 검사기는 테스트 데이터 프리셋을 생성, 저장, 삭제하는 기능과 훈련된 HTM 네트워크의 정확도 테스트를 수행하는 기능을 가진 모듈이다. 테스트 데이터 프리셋은 훈련된 HTM 네트워크의 정확도를 측정하는데 사용할 데이터를 추출 규칙과 함께 명세해둔 것을 말한다. 추출할 데이터는 기준 이미지 수를 정하고 기준 수에 대한 카테고리별 비율로 지정할 수 있으며 추출 규칙만을 기억하고 추출한 데이터는 보관하지 않는다. 즉, 같은 테스트 데이터 프리셋을 사용해도 실제 테스트에 사용되는 데이터는 매번 달라진다.

정확도 검사기는 정확도 테스트를 위해서 데이터 추출 모듈에게 선택된 테스트 데이터 프리셋을 전달하여 테스트 데이터 집합을 추출시키고 추출된 데이터를 넘겨받는다. 이 후, NuPIC을 호출하여 테스트할 훈련된 HTM 네트워크를 불러오고 테스트 데이터를 전달하여 정확도 테스트를 수행한다. 정확도 테스트가 끝나면 결과를 학습 정보 저장소에 저장한다. 매번 테스트를 수행할 때마다 테스트 수행 정보와 결과가 학습 정보 저장소에 저장된다.

4.2.9 NuPIC 제어기와 NuPIC

NuPIC 제어기는 훈련 및 테스트 과정에 사용되는 모듈이 NuPIC을 호출해야 하는 경우에 해당 작업을 대신 처리한다. 각 모듈은 NuPIC 호출이 필요할 때 NuPIC이 아닌 NuPIC 제어기에 요청을 하고, NuPIC 제어기는 요청받은 내용을 분석하여 NuPIC을 호출하여 처리한 후 결과를 호출한 모듈에게 전달한다.

(10) 결과 비교기

결과 비교기는 학습 정보 저장소로부터 테스트 결과를 읽어 비교한다. 결과 비교는 둘 또는 그 이상의 훈련된 HTM 네트워크의 정확도를 비교할 수 있다. 이때 비교 대상이 되는 HTM 네트워크의 훈련시 사용된 카테고리 프리셋이 다를 경우 각 카테고리 프리

셋의 카테고리(양볼 유형) 구성이 다를 수 있기 때문에 비교가 불가능하다. 비교 가능한 대상은 하나의 카테고리 프리셋으로부터 생성된 HTM 네트워크여야 하며, 카테고리 프리셋에 종속된 다른 데이터 프리셋이나 파라미터 프리셋에 의해 훈련된 HTM 네트워크의 정확도를 비교할 수 있다.

4.2.11 HTM 네트워크 배포기와 HTM 네트워크 저장소

HTM 네트워크 배포기는 결과 비교 모듈을 통해 정확도를 비교하여 훈련된 HTM 네트워크 중 최적이라 판단한 HTM 네트워크를 사용자가 선택하면 HTM 네트워크 저장소로부터 실제 양볼 판정 시스템이 훈련된 HTM 네트워크를 사용할 수 있도록 배포한다. HTM 네트워크 저장소는 훈련 시스템에서 만들어진 훈련되었거나 훈련되지 않은 HTM 네트워크들을 임시로 저장하는 공간이다. 하나의 제품 인식에 사용될 훈련된 HTM 네트워크를 선택하여 배포하기 전까지 모든 HTM 네트워크를 담고 있는 공간이며 보관된 HTM 네트워크를 관리한다.

4.2.12 학습 정보 저장소

학습 정보 저장소는 학습 과정에서 발생하는 모든 데이터를 총체적으로 저장하고 관리하며 훈련 시스템의 모듈이 요청시 해당 정보를 제공해 준다. 저장되는 정보중 대표적인 것들은 카테고리 프리셋 명, 카테고리 프리셋의 세부 설정, 데이터 프리셋 명, 데이터 프리셋 설정, 파라미터 프리셋 명, 파라미터 프리셋 세부 설정, 훈련된 HTM 네트워크 정확도 테스트 결과 등이 있다.

5. 시스템 구현

여기서는 구현된 훈련 시스템으로 부품 이미지를 사용하여 HTM 네트워크를 훈련시키고 정확도를 테스트한 후 최적의 정확도를 가진 훈련된 HTM 네트워크를 HTM 기반의 부품 이미지 인식 시스템으로 배포하는 과정을 서술한다.

5.1 훈련 시나리오

훈련시킬 대상 부품은 TM Lock이며 TM Lock의 모양은 그림 3과 같다. TM Lock 부품에 대하여 양볼

유형과 불량품 유형으로 구분하여 훈련시키는데 양품 유형은 그림 4와 같이 양품의 앞면과 뒷면으로 구성된 두 가지 유형으로 각각 good_f, good_r라고 이름 붙인다. 불량품의 유형은 그림 5과 같이 두 가지 유형에 대해 각각 앞면과 뒷면으로 나누어 총 네 가지 유형으로 구성되고 각각 bad_1_f, bad_1_r, bad_2_f, bad_2_r이라고 이름 붙인다. 이러한 유형을 바탕으로 훈련용 데이터 집합과 테스트 데이터 집합을 구성한다.

훈련은 다음과 같은 순서로 진행한다.

1. 제품 정보(부품번호, 부품명) 입력
2. 카테고리 프리셋 생성
3. 데이터 프리셋 생성
4. 파라미터 프리셋 생성
5. 1차 훈련 수행
6. 1차 훈련으로 생성된 HTM 네트워크의 정확도 테스트
7. 설정 단계로 돌아가 파라미터 프리셋 변경
8. 변경된 파라미터 프리셋을 이용한 2차 훈련 수행
9. 2차 훈련으로 생성된 HTM 네트워크의 정확도 테스트
10. 1, 2차 훈련을 통해 생성된 HTM 네트워크간 성능 비교
11. 비교 결과 우수한 HTM 네트워크 배포

5.2 훈련 과정

훈련 대상인 TM Lock에 대한 HTM 네트워크를 생성하고자 할 때는 먼저 부품에 대한 정보를 입력한다. 부품의 기본 정보를 입력한 다음 카테고리 프리셋을 만든다. 하나의 카테고리 프리셋에는 부품에 대한 양품 유형과 불량품 유형의 이름을 입력하고 각 유형에 해당하는 훈련용 이미지 집합과 테스트용 이미지 집합의 경로를 설정한다. 그림 6은 이 단계를 거쳐서 카테고리 프리셋을 설정하는 과정이다.

카테고리 프리셋을 생성한 다음에는 훈련시킬 데이터를 어떻게 추출할지를 결정하는 트레이닝 데이터 프리셋을 생성한다. 트레이닝 데이터 프리셋은 카테고리 프리셋에서 지정한 경로에서 추출하고자 하는 이미지 수를 지정하거나 기준 이미지의 수와 카테고리별 비율을 지정하여 훈련시킬 데이터를 추출할 수 있다. 먼저 그림 7과 같이 트레이닝 데이터 프리셋 이름을 지정한 후, 다음 인터페이스에서 이미지 수나 비율을 선택하면 그림 8과 같이 데이터를 추출하는 작업을 수행한다. 트레이닝 데이터 프리셋 설정이 완료되면 파라미터 프리셋을 설정한다. 파라미터 프리셋 설정 화면은 그림 9와 같다.

파라미터 프리셋까지 설정을 완료하면 ‘트레이닝 시작’ 버튼이 활성화 되고 버튼을 누르면 그림 10와

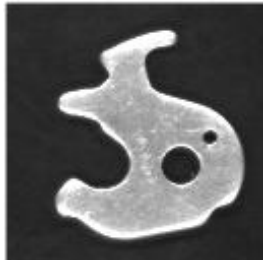
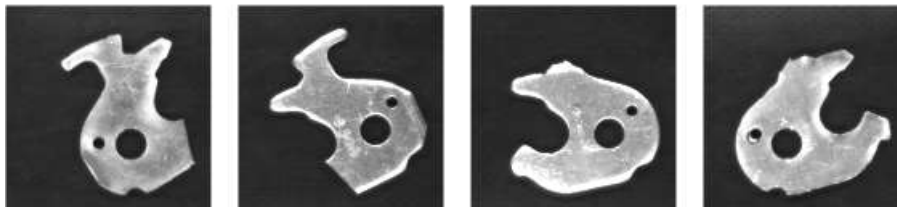


그림 3. TM Lock



good_f good_r

그림 4. TM Lock의 양품 유형과 명칭



bad_1_r

bad_1_f

bad_2_f

bad_2_r

그림 5. TM Lock의 불량품 유형과 명칭

같이 훈련 과정이 진행된다. 훈련 과정은 먼저 훈련 되지 않은 HTM 네트워크가 생성되고 훈련 데이터를 사용하여 훈련을 수행한 후 훈련된 HTM 네트워크가 생성된다.

트레이닝 데이터 테스트는 훈련된 HTM 네트워크의 정확도를 훈련 과정에 사용된 데이터를 통해 측정한다. 진행 과정은 그림 11과 같이 나타나며 테스트 결과는 훈련용 데이터 각각에 대해서 결과를

보여준다. 테스트 결과는 오류가 있는 결과만 볼 수도 있고 그림 12와 같이 전체 보기를 할 수 있다. 그 후 최종적으로 카테고리별 정확도를 그림 13과 같이 백분율로 표현한다.

테스트 데이터 테스트는 테스트용 데이터 집합에서 실제 테스트에 사용될 집합을 추출하여 추출된 집합에 대한 정확도를 측정하는 것이다. 테스트 데이터 테스트를 위해서 그림 14와 같이 테스트 데이터 프리

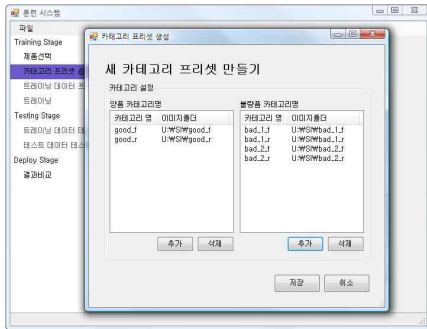


그림 6. 카테고리 프리셋 설정

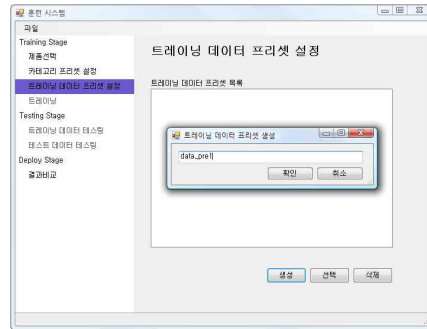


그림 7. 트레이닝 데이터 프리셋 명칭 입력

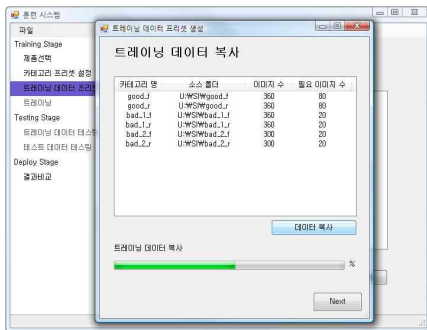


그림 8. 훈련 데이터 집합으로부터 실 훈련 데이터 추출

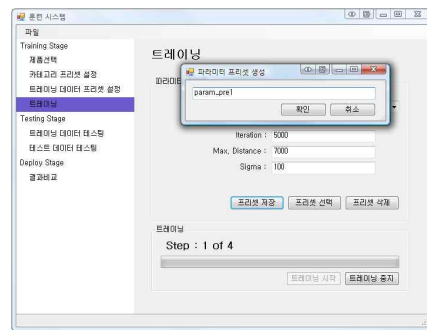


그림 9. 파라미터 프리셋 입력

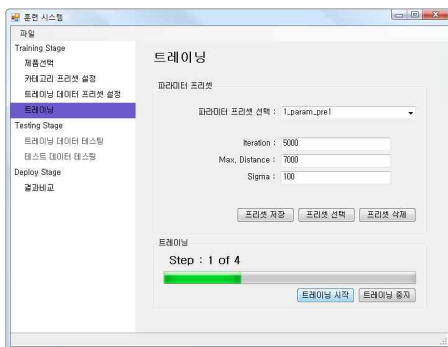


그림 10. 훈련 진행 과정

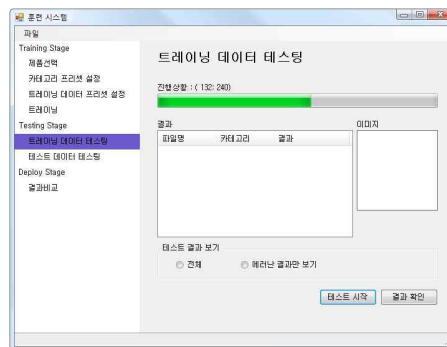


그림 11. 훈련데이터를 이용한 정확도 테스트

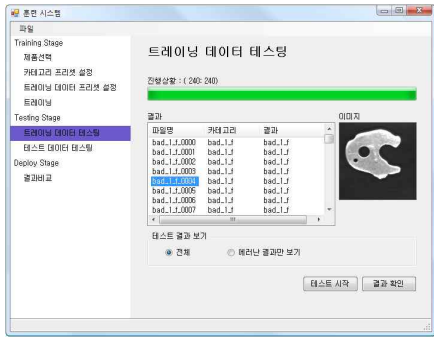


그림 12. 테스트 결과 전체 보기

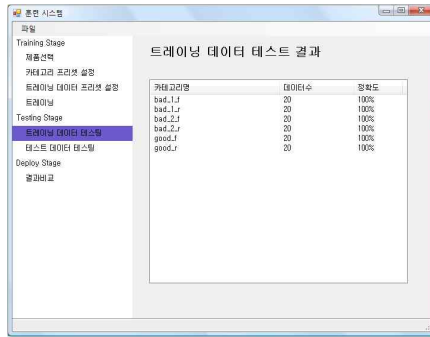


그림 13. 카테고리별 정확도 (백분율)

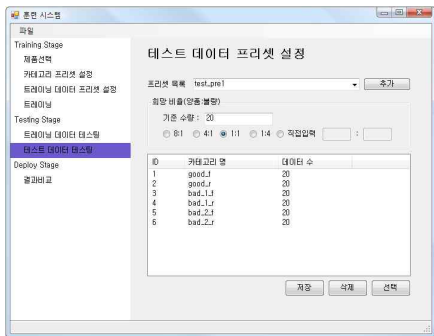


그림 14. 테스트 데이터 집합에서 실 테스트 데이터 추출 방법 지정

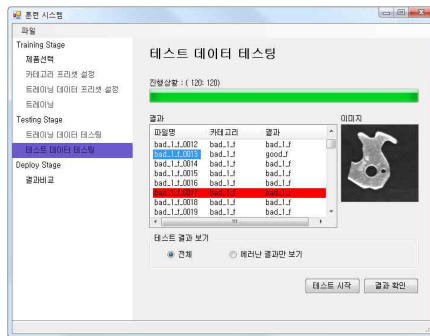


그림 15. 테스트 결과 전체 보기

셋을 생성한다. 테스트 데이터 프리셋은 카테고리별 테스트 데이터 추출 방식을 지정한다. 추출 방법은 그림 14처럼 카테고리별 비율로 추출할 수 있다.

데이터 추출을 완료하면 실제 정확도 테스트 과정이 진행된다. 테스트가 완료되면 테스트 데이터의 개별 결과를 보여주는데 오류가 있는 결과만을 선택해서 볼 수 있고, 그림 15와 같이 모든 결과를 볼 수 있다. '결과 확인' 버튼을 누르면 그림 16과 같이 카테고리에 대한 정확도를 백분율로 표시한다.

이로써 하나의 훈련된 HTM 네트워크를 생성하고 정확도 테스트 과정이 끝난다. 다음으로 훈련 단계의 파라미터 프리셋만 변경하여 훈련 후 첫 번째 훈련과 두 번째 훈련의 결과를 비교하여 우수한 HTM 네트워크를 선택하여 배포하는 과정을 서술한다.

본 훈련 도구는 피드백 기능을 지원하기 때문에 훈련 과정의 첫 단계가 아닌 파라미터 값을 설정할 수 있는 단계로 가서 파라미터 값만 변경 후 훈련을 수행할 수 있다. 그림 17처럼 훈련 단계의 '트레이닝'을 선택하여 파라미터 값을 변경하고 파라미터 프리

셋을 저장하는 것으로 다른 설정은 동일하게 유지한 상태로 원하는 설정만 변경할 수 있다. 이전의 훈련 과정과 마찬가지로 트레이닝 과정을 거치면 훈련된 HTM 네트워크가 생성되고 같은 테스트 데이터 프리셋으로 테스트를 거치면 그림 18과 같이 결과를 보여준다.

이로써 두 번째 훈련을 종료하고 두 개의 훈련된 HTM 네트워크의 정확도를 비교하여 최적의 훈련된

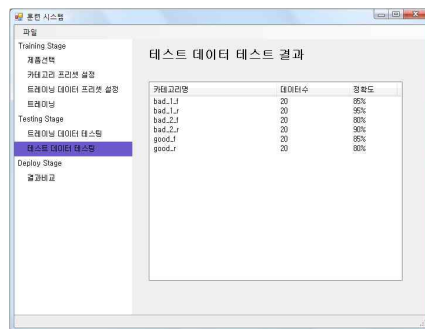


그림 16. 테스트 데이터 카테고리별 결과 (백분율)

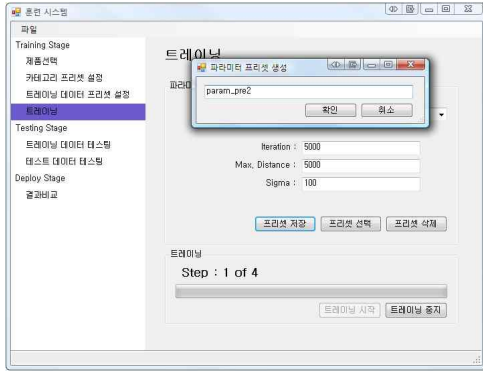


그림 17. 파라미터 프리셋 변경

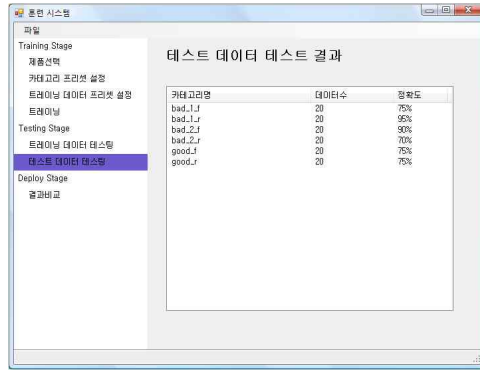


그림 18. 2차 테스트 결과(백분율)

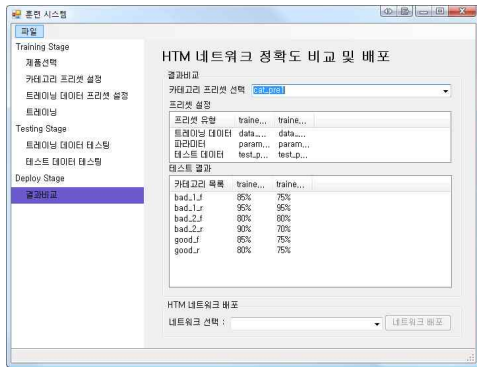


그림 19. 훈련된 HTM 네트워크간의 정확도 비교

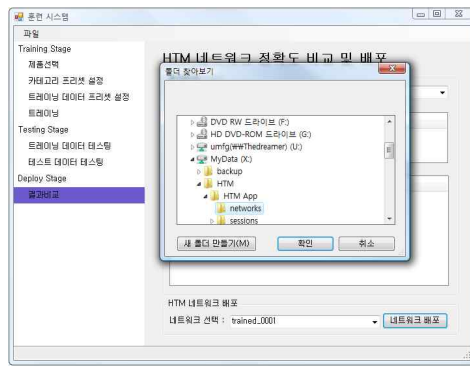


그림 20. 훈련된 HTM 네트워크 배포

HTM 네트워크를 배포하는 과정을 서술한다. HTM 네트워크의 정확도 비교와 배포는 배포 단계의 '결과 비교'에서 할 수 있다. 부품에 대한 카테고리 프리셋을 선택하면 그림 19와 같이 해당 카테고리 프리셋에서 훈련을 위해 선택한 데이터 프리셋, 파라미터 프리셋, 테스트 데이터 프리셋의 종류를 보여주고 카테고리 프리셋에 정의된 카테고리별 정확도를 보여준다.

결과 중 적절한 인식률을 지닌 HTM 네트워크가 없을 경우는 설정 - 훈련 - 테스트를 반복할 수 있으며, 최적의 훈련된 HTM 네트워크로 판단된 것이 있을 경우는 원하는 훈련된 HTM 네트워크를 선택하여 그림 20과 같이 HTM 기반의 부품 이미지 인식 응용이 위치한 경로를 선택하여 해당 경로에 HTM 네트워크를 배포하는 것으로 부품 하나에 대한 훈련이 종료된다.

6. 평 가

본 논문에서는 제시한 훈련 시스템이 개발되기 이

전에는 많은 반복된 수작업과 사용의 편의성 문제 때문에 부품의 양불량을 판정하는 시스템을 구축함에 있어서 많은 시간이 소비되었다. 한편으로는 본 시스템 사용자는 현장 부품생산기술자라는 점을 감안해서 설계한 결과, 다음의 성능으로 인하여 시스템 개발과정을 크게 단순화시켜서 개발 시간을 크게 단축할 수 있었다.

첫째, 네트워크 생성과정에서 본 훈련 시스템은 기존에 제안된 도구와는 다르게 부품 이미지 인식분야를 목표로 개발되었기 때문에 부품 이미지를 이용한 양불여부 판정에 적합한 HTM 네트워크의 구조를 기본적으로 제공한다. 제공되는 HTM 네트워크 구조는 많은 테스트를 거쳐 부품 이미지 인식에 가장 최적화되었다. 앞에서 언급한 바와 같이 HTM 네트워크 구조 생성은 전문기술을 요하기 때문에 이 특징은 HTM을 모르는 현장 기술자에게 가장 중요한 기능이다.

둘째, 훈련절차 중에서 제품 카테고리를 설정할 때 기존 도구들은 사용자가 직접 카테고리를 설정하

여 그 카테고리에 부품이미지 파일을 저장해야 하지만, 본 시스템에서는 카테고리 프리셋 생성을 지원하여 해당 카테고리에 부품이미지를 자동으로 랜덤으로 저장한다. 또한 훈련된 네트워크의 정확도를 검증한 후 파라미터의 변경을 통하여 훈련절차를 반복하게 되는데, 이때 기존 도구들은 매번 파라미터를 새로 입력해야 하는 반면에 본 시스템은 파라미터 프리셋에서 선택만 하면 되므로 작업과정을 크게 단순화시킨다. 즉, 훈련된 HTM 네트워크의 정확도를 높이기 위해서는 설정 변경-훈련-테스트 과정이 반복되는데, 본 시스템은 사용자가 입력한 HTM 네트워크 훈련 설정을 기억하여 훈련된 HTM 네트워크의 테스트 결과가 만족할 수준의 정확도를 보여주지 않을 경우는 사용자가 원하는 설정 단계로 되돌아가 그 단계의 파라미터 설정만 변경하여 훈련을 진행할 수 있는 피드백 기능을 지원한다. 따라서 사용자는 매번 전체적인 훈련 과정을 반복할 필요 없이 원하는 설정만을 수정하여 다시 훈련을 수행할 수 있다. 이

는 훈련의 과정을 크게 단순화 시킨다.

셋째, HTM 훈련결과 평가과정에서 한 번의 설정-훈련 과정이 진행될 때마다 하나의 HTM 네트워크가 생성되고 정확도 테스트가 수행된다. 정확도를 비교하기 위해서는 결과를 저장해야 한다. 본 훈련 시스템은 반복되는 훈련과정 중에서 HTM 네트워크 생성 조건, 훈련된 HTM 네트워크 파일, 각 HTM 네트워크들의 정확도 테스트 결과를 저장하고 필요시 HTM 네트워크들의 정확도를 쉽게 비교할 수 있는 반면에 기존 시스템은 이 과정이 완전히 수작업으로 이루어진다. 표 1은 이를 요약하여 훈련절차에 따른 자동화 수준과 편의성에 따른 성능을 비교 분석한 것이다.

7. 결론 및 향후 과제

본 논문에서는 생산 현장에서 부품의 이미지로 불량 여부를 판단할 수 있도록 훈련된 HTM 네트워크를 생성할 수 있는 부품 이미지 인식을 위한 훈련 시스템

표 1. 훈련절차에 따른 성능 평가

		RunExperiment	Vitamin d toolkit	본 훈련 시스템
네트워크 생성과정		HTM네트워크 구조 설계 + HTM 네트워크 구현(프로그래밍 방식) + 테스트 + 디버깅 모든 과정 수작업	HTM네트워크 구조 설계 + HTM 네트워크 구현(GUI방식) + 테스트 + 디버깅 모든 과정 수작업	네트워크가 이미 내장되어 있음.
훈련 절차	제품카테고리설정	카테고리 설정을 사용자가 직접 입력	카테고리 설정을 사용자가 직접 입력	카테고리 프리셋 기반의 반자동화
	훈련파라미터설정	사용자가 params.py 파일 내용을 매번 수정	GUI를 통해서 매번 수동으로 입력	파라미터 프리셋에서 선택. (새로운 프리셋도 생성 가능)
	네트워크 훈련수행 명령	텍스트 명령어	GUI	GUI
훈련 결과 평가 과정	훈련결과 의 표현	report.txt 및 로그파일에 저장. 사용자는 이 파일들을 읽어서 분석함	훈련결과를 화면에 제시.	훈련결과를 화면에 제시.
	훈련결과 의 관리	카테고리/파라미터 변경에 따른 모든 훈련결과 의 수동 저장	카테고리/파라미터 변경에 따른 모든 훈련결과 의 수동 저장	카테고리/파라미터 프리셋에 따른 훈련결과를 자동으로 데이터베이스로 관리
	훈련된 네트워크 저장	가장 좋은 성능을 가진 HTM네트워크를 수동 선택, 이미 훈련된 네트워크를 수동으로 저장	가장 좋은 성능을 가진 HTM 네트워크를 수동 선택, 이미 훈련된 네트워크를 수동으로 저장	데이터베이스를 통해서 자동으로 관리. 최선의 네트워크를 자동 선택하여 배포

을 제시하였다. 본 훈련 시스템을 활용하여 현장 기술자가 직접 이미지만으로 제품의 불량 여부를 판단하는 HTM 네트워크를 생성할 수 있어 일회 도입으로 모든 부품에 적용할 수 있으며 도입 이후에는 전문가가 별도로 필요하지 않아 비용 측면에서도 유리하다.

HTM 기반의 부품 이미지 양불 여부 판정 시스템과 본 훈련 시스템을 중소기업의 생산업체에서 도입할 경우는 적은 비용으로 생산하는 모든 부품에 적용 가능하며 높은 인식률을 지닌 검사 시스템을 갖출 수 있을 것으로 예상된다.

본 훈련 시스템은 HTM의 이미지 인식 능력을 활용하여 제품 이미지로 불량여부를 판단하기 위한 HTM 네트워크를 훈련시키는 것을 목표로 하였다. 하지만 실제 공장에서 생산되는 제품의 불량 여부는 프레스 기계가 제품을 생산할 때 가해지는 온도나 압력과 같은 요소에도 많은 영향을 받는다. 향후 HTM 응용 분야의 발전 방향은 최종 생산된 제품에 대한 불량 여부 판단뿐만 아니라 제품 생산에 영향을 주는 여러 가지 요소들을 분류하고 그 데이터를 학습하여 생산 공정을 전반적으로 관리해 주는 시스템이다. 이를 위해서는 이미지뿐만 아니라 온도 센서나 압력 센서 등 여러 센서들로부터 획득한 데이터를 종합적으로 학습할 수 있는 HTM 네트워크 훈련 시스템이 필요하다.

참 고 문 헌

- [1] 이상학, 서명호, 정태충, “이미지 프로세싱을 이용한 자동 너트 검사 장비 개발,” 한국정보처리학회논문지, 제11-A권 제4호, pp. 235-242, 2004. 4.
- [2] Jeff Hawkins, “On Intelligence”, <http://www.onintelligence.com>, 2005.
- [3] Jeff Hawkins and Dileep George, “Hierarchical Temporal Memory,” http://numenta.com/Numenta_HTM_Concepts.pdf, March 2007.
- [4] Numenta, Inc., “Numenta Platform for Intelligent Computing,” http://numenta.com/for-developers/software/pdf/nupic_plug_guide.pdf, June 2008.
- [5] 배선갑, 이대한, 조건화, 남해보, 성기원, 배종민, 강현석, “중소 제조업을 위한 HTM 기반의 부품 이미지 인식 시스템의 개발,” 정보처리학회논문지D 제16-D권 제4호, pp. 613-620, 2009.8.
- [6] Numenta, Inc., “Vision Framework Guide,” July 2008.
- [7] Vitamin D, Inc., “Vitamin D Toolit Reference Guide,” [http://www.vitamindinc.com/downloads/Vitamin D Toolkit Reference Guide.pdf](http://www.vitamindinc.com/downloads/Vitamin_D_Toolkit_Reference_Guide.pdf) June 2008.
- [8] Dileep George and Bobby Jaros, “The HTM Learning Algorithms,” http://numenta.com/for-developers/education/Numenta_HTM_Learning_Algos.pdf March 2007.
- [9] Vitamin D, Inc., “Advanced NuPIC Programming,” http://www.numenta.com/for-developers/software/pdf/nupic_prog_guide.pdf September 2008.
- [10] Numenta, Inc., “Getting Started with NuPIC,” http://numenta.com/for-developers/software/pdf/nupic_gettingstarted.pdf September 2008.



이 대 한

2004년 경상대학교 컴퓨터과학과 졸업(학사)
2007년~2009년 경상대학교 대학원 컴퓨터과학과 재학(석사)
2009년~현재 경상대학교 병원

관심분야: XML, 데이터베이스 통합, 기계학습



강 현 석

1981년 동국대학교 전자계산학과 졸업(학사)
1983년 서울대학교 대학원 전산학과 졸업(석사)
1989년 서울대학교 대학원 전산학과 졸업(박사)

1981년~1984년 2월 한국전자통신연구원 연구원
1984년 3월~1993년 2월 전북대학교 전임강사, 부교수, 교수

1993년 3월~현재 경상대학교 컴퓨터과학과 교수
관심분야: 멀티미디어, 내장형 데이터베이스, 지능시스템



배 선 갑

1987년 경북대학교 물리학과 졸업(학사)
2004년 경상대학교 대학원 컴퓨터과학과 졸업(석사)
2006년~현재 경상대학교 대학원 컴퓨터과학과 박사과정

관심 분야: 패턴인식, 지능시스템, 내장형 데이터베이스



배 종 민

1980년 서울대학교 수학교육과 졸업(학사)
1983년 서울대학교 대학원 계산통계학과 졸업(석사)
1995년 서울대학교 대학원 계산통계학과 졸업(박사)

1982년~1984년 한국전자통신연구소 연구원
1997년~1998년 Virginia Tech. 객원연구원
1984년~현재 경상대학교 컴퓨터과학부 교수
관심분야: XML, 데이터베이스 통합, 기계학습



서 대 호

2009년 경상대학교 컴퓨터과학과 졸업(학사)
2010년 경상대학교 대학원 컴퓨터과학과
관심 분야: XML, 기계학습, 데이터베이스 통합