

# 영상처리 알고리즘 개발을 위한 소프트웨어형상관리시스템

이 정 현<sup>†</sup> · 채 옥 삼<sup>††</sup>

## 요 약

디지털기기 개발에 있어서 소프트웨어의 중요성이 높아지고, 규모가 커짐에 따라 소프트웨어의 위기를 해결하기 위한 소프트웨어형상관리의 필요성이 증가하고 있다. 그러나 일반적인 소프트웨어형상관리 시스템들은 영상처리 알고리즘 개발 환경의 특성 및 특징들을 수용하기에 부족하다. 영상처리 알고리즘 개발 환경은 컴파일러와 같은 기본적인 개발 도구 외에 컴포넌트(나 라이브러리)들을 인터랙티브하게 조합하고 테스트 할 수 있는 시뮬레이션 환경을 통하여 테스트와 분석을 반복하면서 개발하는 특징을 가지고 있다. 뿐만 아니라, 시뮬레이션 과정 중에는 유사한 기능을 가진 알고리즘 중에 적합한 알고리즘을 빠르고 효과적으로 찾기 위한 방법이 필요하다. 본 논문에서는 이러한 영상처리 알고리즘 개발 환경에서 사용하는 시뮬레이션 도구를 체계적으로 지원해줄 수 있는 비주얼 워크스페이스 기반의 소프트웨어형상관리 방법과 이를 수용할 수 있는 소프트웨어형상관리 시스템을 제안한다. 또한, 개발된 시스템을 실제 시뮬레이션 도구와 연계하여 소프트웨어형상관리 시스템으로써의 적합성을 확인한다.

## The software configuration management system for image processing algorithm development

Jeong-Heon Lee<sup>†</sup> · Ok-Sam Chae<sup>††</sup>

### ABSTRACT

The importance of software is getting high in development of the digital device (digital camcorder, digital camera, mp3 player, ....). And because the size of software becomes larger and complicated, the necessity of software configuration management (to solves a software crisis) is increased. The general software configuration management system shows lack of the property and features of software development environment for image processing algorithm due to its wide range to be covered. Image processing algorithm development environment has properties like repetitive analysis and simulation using visual programming environment where, beside support of elementary development functions, component(or library) can be combined and tested interactively. Moreover, the method to look fast and effectively for component having similar function is required. In this paper, we present the system which supports the software configuration management method for a simulation tool and the property in the visual programming environment. And we relate our system to real simulation tool so as to check its ability as the software configuration management system for image processing algorithm development environment.

**키워드 :** 소프트웨어형상관리시스템(Software Configuration Management System), 영상처리 알고리즘 개발환경(Image Processing Algorithm Development Environment), 비주얼 워크스페이스 기반 소프트웨어형상관리(Visual Workspace Based SCM), 영상처리 알고리즘 개발 환경(Image Processing Algorithm Development Environment)

### 1. 서 론

네트워크와 하드웨어의 기술은 소프트웨어에 대한 컴퓨터 사용자의 기대치를 더욱 키웠으며 새로운 소프트웨어 기능에 대한 요구도 더욱 증가하게 하였다. 이러한 상황에서 소프트웨어 크기는 보다 급속히 확대되고 있으며 개발 도구의 발전으로 인하여 소프트웨어 개발 주기도 짧아지고 있다. 최근에는 다수의 개발자가 동시에 개발에 참여하는

복잡하고 규모가 큰 소프트웨어를 체계적으로 개발하고 관리하여 새로운 버전의 소프트웨어를 보다 효율적으로 개발하기 위한 연구들이 활발히 진행되고 있다[1-3].

이러한 연구 중의 하나가 바로 소프트웨어형상관리(SCM : Software Configuration Management) 기술이다. SCM은 소프트웨어의 전체 생명주기를 통해 형상(Configuration)을 인식하여 체계적으로 변경 사항을 관리하고, 소프트웨어의 추적 용이성과 통일성을 유지하기 위한 활동을 가리킨다. 이러한 SCM 활동들을 시스템적으로 지원해주기 위해 개발된 것이 바로 SCM 시스템이다 [4-7, 11-12].

<sup>†</sup> 정 회 원 : 경희대학교 대학원 전자정보학부

<sup>††</sup> 정 회 원 : 경희대학교 전자정보학부 교수, 교신저자

논문접수 : 2004년 8월 20일, 심사완료 : 2004년 11월 30일

SCM 시스템은 초기에는 규모가 큰 일반 업무용 소프트웨어 개발 분야를 지원하기 위해서 주로 연구되었으나, 최근 들어 영상 센서를 기본으로 장착한 캠코더, 카메라, 핸드폰과 같은 디지털 기기가 일반화되고 이들 장치를 지원하는 소프트웨어의 규모가 커지고 개발주기가 짧아지면서 영상처리 기술을 포함하는 디지털 신호처리 기술 개발을 위한 SCM 시스템의 필요성이 대두되고 있다[8-9, 20, 23]. 그러나 기존의 일반적인 SCM 시스템들은 업무절차(Business Process)용 소프트웨어 개발 도구에 적합하게 개발되어, 영상처리 알고리즘 개발환경에 적용하기에는 부족한 면이 많다[13-15].

영상처리 알고리즘 개발 환경은 업무절차용 소프트웨어 개발 환경과 다음과 같은 기본적인 차이를 가지고 있다 [16-18, 20-25].

- ① 업무절차용 소프트웨어의 개발에서는 설계 시 목표로 했던 결과 값과 동일하여야 개발이 종료되지만, 영상처리용 소프트웨어의 경우 설계 시 목표로 했던 결과 값과 근사한 결과에서 종료되는 경우가 많고 그럼으로 인하여 개발 종료시점의 예측도 어렵다.
- ② 업무절차 컴포넌트들은 대부분 유일한 기능을 가지므로 키워드 검색만으로 컴포넌트 저장소에서 원하는 기능의 컴포넌트를 쉽게 찾을 수 있으나, 영상처리 컴포넌트들의 경우 동일한 기능을 수행하는 컴포넌트들이 다수 존재하기 때문에 단순히 키워드 검색만으로는 원하는 기능의 적합한 컴포넌트를 찾기 어렵다.
- ③ 업무절차 컴포넌트 개발 시에는 개발 중간의 산출물이 사용되는 경우가 없지만, 영상처리 컴포넌트 개발 시에는 개발 중간의 산출물이 설계 시의 환경과는 다른 곳에서는 효과적인 컴포넌트를 사용되는 경우들이 존재한다.
- ④ 업무절차용 소프트웨어 개발에서는 주로 라이브러리(혹은 컴포넌트)와 컴파일러만을 이용하여 개발 진행되지만, 영상처리용 소프트웨어의 경우 개발 과정 중에 반복적인 시뮬레이션과 분석 단계를 가지고 있어 컴포넌트(혹은 라이브러리)를 쉽게 조합하여 아이디어를 확인하고 결과를 확인해 볼 수 있는 시뮬레이션 환경(영상처리 알고리즘 라이브러리와 그 라이브러리를 이용하여 새로운 알고리즘을 생성하거나 응용시스템을 개발할 수 있는 비주얼 프로그래밍 환경을 갖추고 있다.)을 병행 사용한다.
- ⑤ 업무절차 컴포넌트와는 달리 영상처리 컴포넌트는 사용되는 위치에 따라서 성능과 결과가 달라지는 경우가 많다. 또한, 컴포넌트를 사용하는 용도가 다양하게 생성될 가능성이 높기 때문에 이러한 정보를 수용하여 개발자 지식을 저장할 수 있는 방법이 필요하다.

기존의 SCM 시스템들[26-29]은 전문화된 영상처리 분야와 개발 환경의 특성을 효과적으로 지원해주지 못하고 있어 SCM 활동의 효과를 반감시키고 있다. 영상처리 알고리즘 개발 환경에 적합한 SCM 시스템이 되기 위해서는 비주얼 워크스페이스(비주얼 프로그래밍 환경의 작업영역) 기반의 형상관리 기능과 컴포넌트 개발과정 중에 발생하는 중간산출물이 최종 산출물로 사용될 수 있다는 점을 고려하여야 하고, 원하는 컴포넌트를 쉽게 찾을 수 있는 기능이 지원되어야 한다.

본 논문에서는 이러한 영상처리 알고리즘 개발환경의 특성을 고려하여 비주얼 워크스페이스 기반의 하이퍼링크 검색이 가능한 SCM 시스템을 개발하고, 실제 사용 중인 시뮬레이션 환경과의 인터페이스를 통하여 영상처리 알고리즘 개발환경에서의 적합성을 확인하였다.

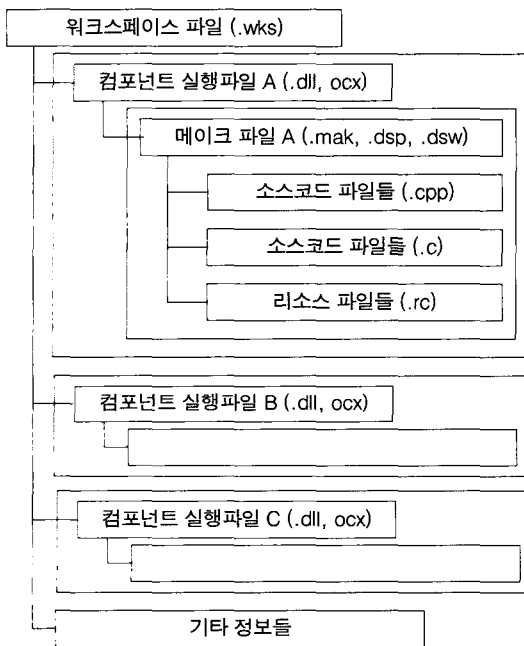
## 2. 영상처리 알고리즘 개발 환경을 위한 소프트웨어 형상관리 구조의 설계

서론에서 전문화된 영상처리 알고리즘 개발환경의 특징을 지원하기 위한 SCM 시스템을 개발하기 위해서는 다음과 같은 세부 사항을 고려하여야 한다.

- ① 업무절차용 SCM 시스템과 가진 소스 코드 기반의 SCM 기능을 포함하여야 하며, 영상처리 알고리즘 개발환경에서 사용되는 시뮬레이션 환경의 작업영역인 비주얼 워크스페이스에 적합한 SCM 기능을 제공하여야 한다. 즉, 일반적인 소스코드 파일 중심의 SCM 기능인 버전관리, 리빌드관리, 워크스페이스관리 등을 기본적으로 지원해 줄 뿐만 아니라 비주얼 워크스페이스 중심으로도 같은 기능이 지원되어야 한다.
- ② 시뮬레이션 환경의 작업 파일인 비주얼 워크스페이스를 위한 디스플레이 기능이 지원되어야 한다. 이러한 비주얼 한 디스플레이 기능의 지원은 컴포넌트 및 응용 소프트웨어에 대한 관리 및 이해력을 증가시킬 수 있어 효과적인 재사용을 유도할 수 있다.
- ③ 개발과정 중에 발생하는 중간산출물을 개발에 사용되는 베이스라인과는 다른 베이스라인의 최종 산출물로 분기할 수 있어야 한다. 영상처리 분야의 특성상 동작환경에 따라 중간산출물로 나온 결과물들이 다른 환경에서는 최적의 솔루션으로 적용되는 경우가 다수 발생하기 때문이다.
- ④ 영상처리 알고리즘 개발 시에는 최적의 솔루션을 얻기 위하여 실제 데이터를 이용한 반복적인 시뮬레이션과 분석과정을 통하게 되고 이를 통하여 적합한 컴포넌트를 선택 사용하거나 적합한 기능을 개발하게 된다. 이러한 반복되는 개발과정의 시간을 줄여주기 위해서 관련된 정보들을 빠르게 검색하고 이동할 수 있는 방법이 필요하다.

2.1 워크스페이스 기반의 형상 자료 저장 구조 설계

전술한 4가지 사항들을 고려하여 영상처리 알고리즘 개발환경에서 시뮬레이션에 기준이 되는 워크스페이스 파일을 기반으로 한 형상관리 방법을 제안하였다. 영상처리 알고리즘 개발 환경의 워크스페이스 파일은 다수의 컴포넌트 정보를 포함하는 응용시스템이나 복잡한 컴포넌트 단위를 구성하기 위해서 사용되는 파일로 특정 목적을 달성하기 위하여 컴포넌트 사이의 관계 정의를 포함하고 있다. 이러한 워크스페이스 파일을 개발 과정 중에서 생성되는 파일들과의 논리적인 관계를 표현한다면 다음 (그림 1)과 같은 관계를 가지고 있다고 볼 수 있다. 워크스페이스 파일은 여러 개의 실제적인 컴포넌트 실행 파일의 정보를 가지고 있지만, 실제로 그 컴포넌트 실행파일을 만들기 위해서는 컴파일러를 위한 메이크파일(makefile)과 실제적인 알고리즘과 인터페이스들이 들어있는 파일들이 있어야 하기 때문에 (그림 1)과 같은 관계가 된다고 볼 수 있다.



(그림 1) 워크스페이스 파일 구성을 위한 파일들간의 논리적인 관계

본 논문에서는 이러한 논리적인 관계를 형상관리 구조에 적용한 워크스페이스 파일 중심의 형상관리 방법에 적용하였다. 형상자료 관리 단위를 계층적으로 구성하여 제일 상층에 워크스페이스를 기준으로 워크스페이스 파일의 히스토리(history) 정보와 히스토리 정보에 보관되어 있는 각 워크스페이스에 포함된 컴포넌트의 세부정보들이 기록되도록 설계하였으며, 일반적인 형상관리 환경들이 가지는 컴포넌트 각각의 세부적인 소스코드 파일의 히스토리 정보를 유지하도록 구성하였다. 제안된 형상관리 자료의 저장 구조

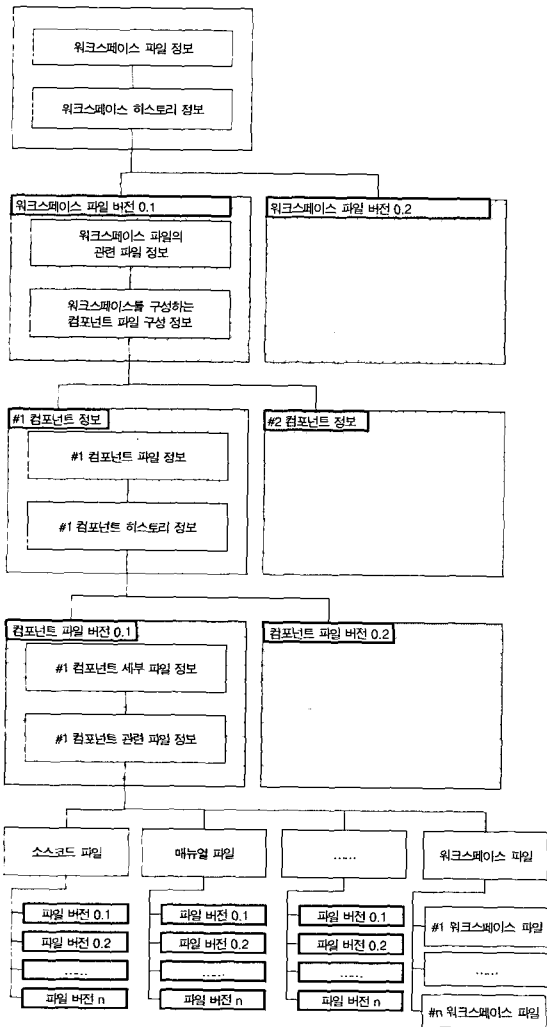
는 (그림 2)와 같다. 제안 시스템은 (그림 2)와 같은 관리 구조를 기반으로 (그림 3)와 같은 전체적인 구성을 이루도록 하였다.

2.2 제안 시스템 전체 구조

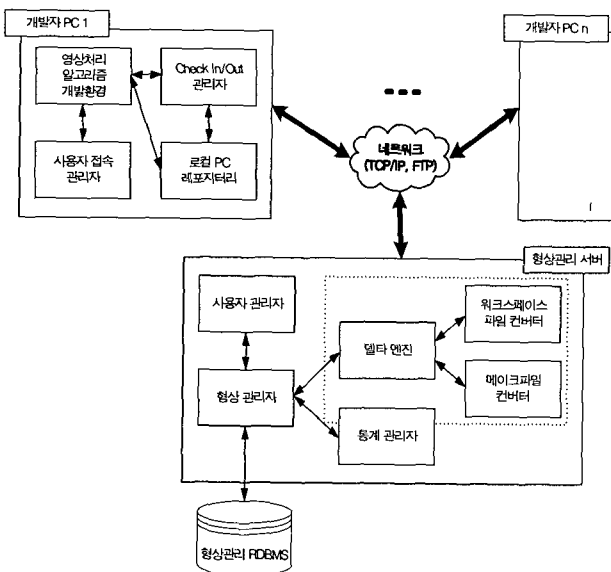
제안 시스템은 형상데이터의 저장 및 복원을 위한 형상관리서버와 영상처리 알고리즘 개발 환경 지원을 위한 클라이언트 부분으로 구성되어 있다. (그림 3)의 하단에 위치하고 있는 형상관리서버 부분은 개발 및 유지보수 과정 중에 생기는 형상을 체계적으로 저장하거나 관리하고, Check In/Out 과정을 통하여 사용자에게 형상 데이터의 분배 및 저장을 담당하며, 개발 및 유지보수 과정에 참여한 사용자들을 관리하기 위한 기능들로 구성된다. 상단에 위치하고 있는 클라이언트 부분은 개발자들이 자신의 PC에서 개발과정 중에 생성되는 모든 형상 데이터를 서버로 다운로드나 업로드 하는 기능을 수행하며, 영상처리 알고리즘 개발 환경의 인터페이스를 제공한다.

(그림 3)의 하단에 있는 비주얼 워크스페이스 파일 컨버터는 시뮬레이션 도구에서 만들어진 비주얼 워크스페이스 파일을 형상관리서버로 임포트(importing)하는 역할을 수행하며, 소프트웨어 개발에 사용하는 상용 컴파일러의 메이크파일(Makefile)이나 프로젝트 파일(Project File)을 임포트하는 것은 프로젝트 파일 컨버터가 담당한다. 형상관리서버에 데이터를 효과적으로 저장하기 위한 압축 엔진인 델타엔진(delta engine)과, 형상관리 서버에 접속할 수 있는 사용자들의 리스트 관리 및 접근 권한 관리를 위해 사용자 관리자를 두었다. 형상관리자는 개발용 PC로부터 Check In되는 모든 파일 및 이벤트 정보를 분석하여 하이퍼링크(hyper link)정보를 생성하며, 사용자 및 날짜 정보와 같이 형상관리 RDBMS에 저장, 복원, 검색하는 역할을 수행한다. 그리고 형상관리서버에 누적된 형상관리 정보에 대한 통계 데이터 생성, 분석 및 관리를 위한 통계관리자와 형상관리 데이터들을 저장하기 위한 형상관리 RDBMS가 있다.

(그림 3)의 상단에 있는 영상처리 알고리즘 개발환경은 영상처리 알고리즘 소프트웨어를 개발할 때 사용하는 도구로 보통 컴포넌트(혹은 라이브러리)를 조합하여 시뮬레이션 할 수 있는 비주얼 프로그래밍 환경을 가지고 있으며, 비주얼 프로그램으로 구성된 비주얼 워크스페이스 파일을 생성한다. 프로그래머가 개발용 PC에 형상관리 서버로부터 Check In/Out 절차를 통하여 다운로드 받은 관련 파일들을 임시로 저장하여 시뮬레이션 도구가 사용할 수 있도록 해주는 저장 장소인 로컬 PC 레포지터리가 있으며, 사용자 접속 관리자는 형상관리서버로부터 데이터 파일을 다운로드/업로드 하기 위해서 형상관리 서버로 접속할 수 있는 인증을 받기 위한 관리자 기능 및 클라이언트 PC에서 접속하는 사용자들의 정보를 관리하는 역할을 수행한다.



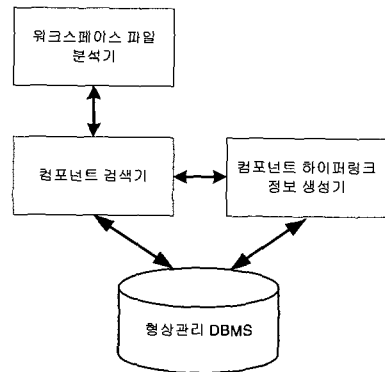
같은 파일의 다른 버전 ... 다른 파일  
 (그림 2) 워크스페이스 기반의 형상 자료 저장 구조



(그림 3) 제안 시스템의 전체 구성

2.3 워크스페이스 기반의 하이퍼링크생성 구조

영상처리 알고리즘 개발 환경과 같이 시뮬레이션을 사용하는 개발에서는 소스코드 외에 비주얼 프로그래밍 환경의 비주얼 워크스페이스 또한, 체계적인 관리가 필요한 중요한 형상 관리의 대상이 된다. 뿐만 아니라 고려사항에서 언급한 바와 같이 개발 단계상에 발생하는 시뮬레이션과 분석 중에 발생하는 개발기간을 단축시켜주기 위하여 컴포넌트를 선택하면 바로 이동하여 관련 정보를 확인할 수 있고 관련 정보들을 통하여 다른 정보로 빠르게 이동할 수 있도록 하기 위하여 컴포넌트에 하이퍼링크(hyperlink) 정보를 지원하도록 하였다.



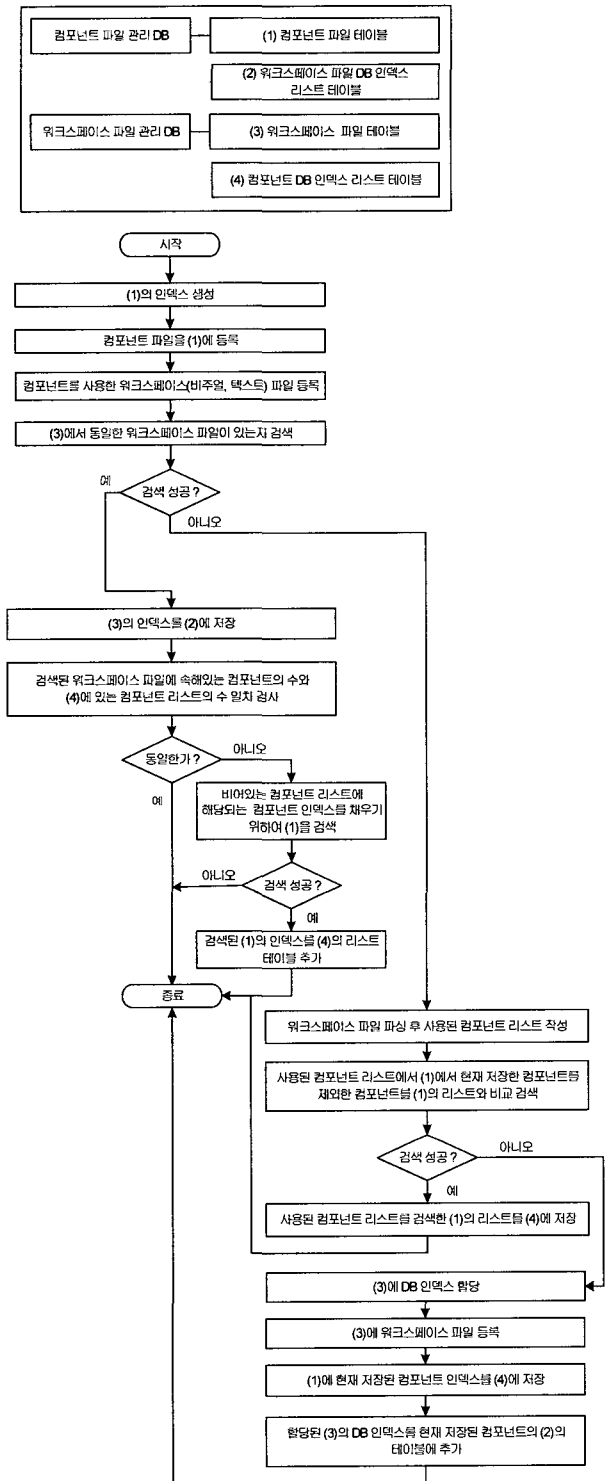
(그림 4) 하이퍼링크 정보 생성 모듈

이러한 하이퍼링크정보의 생성을 위해서 영상처리 알고리즘 개발환경의 비주얼 워크스페이스의 파일에서 사용하고 있는 컴포넌트(혹은 라이브러리)의 위치 정보를 (그림 3)의 형상관리자로부터 얻은 후 하이퍼링크 정보를 구성하여 (그림 2)와 같은 형상데이터 저장 구조에 저장하도록 하였다. (그림 4)는 (그림 3)의 형상관리자 안의 하이퍼링크 정보 생성 모듈을 보여주고 있다. 하이퍼링크 생성 모듈은 비주얼 워크스페이스 파일을 분석하여 형상관리서버에 등록되어 있는 컴포넌트(혹은 라이브러리)를 검색한 후 비주얼 워크스페이스에 사용된 컴포넌트들의 위치 정보를 기반으로 링크정보 생성기를 통해 하이퍼링크 정보를 생성 저장하게 된다. (그림 4)의 하이퍼링크 정보 생성 모듈은 새로운 컴포넌트나 워크스페이스 파일이 등록될 때마다 컴포넌트 사이의 하이퍼링크 정보들이 늘어나게 한다. (그림 5)는 형상관리 환경에 등록되는 각 컴포넌트들 사이의 하이퍼링크 정보를 자동으로 생성하는 절차를 나타내고 있다.

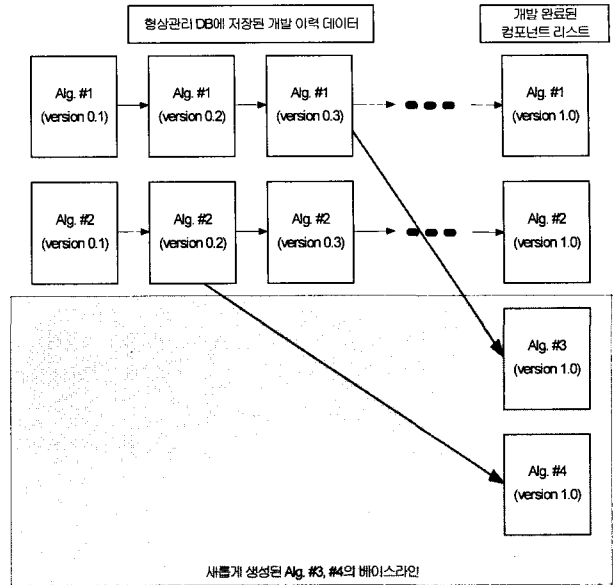
2.4 중간결과물의 새로운 응용시스템의 이용을 위한 최종 결과물로의 브랜치 기능

전술한 바와 같이 영상처리 알고리즘들은 개발도중에 생기는 중간산출물들이 다른 환경에서는 최종결과물로 사용되는 경우들이 많다[16-18]. 이러한 특징 때문에 SCM 시스템은 개발과정 중에 발생하는 중간산출물을 분기하여 새로운 소프트웨어인 최종솔루션으로 등록할 수 있는 구조를 갖도록 하였다. 뿐만 아니라 분기된 소프트웨어가 개발 이

력(history)을 가질 수 있도록 기존 개발이력 정보의 위치 정보를 연결정보로 저장하도록 하였다. 기존의 SCM 시스템들은 새로운 베이스라인(baseline)이나 브랜치(branch)로 관리하는 기능들을 지원하고 있지만 이러한 기능들은 개발의 종료기점이 아닌 개발시작기점(베이스라인 : baseline)으로써 관리하기 때문에 이 같은 경우에는 적용할 수 없다.



(그림 5) 하이퍼링크 정보 자동 생성 절차



(그림 6) 중간산출물의 분기 구조

개발과정 중에 발생하는 중간산출물을 선택적으로 분기하기 위해선 개발에 사용되고 있는 소스코드 파일 및 워크스페이스 파일의 각 리비전(revision)을 선택적으로 분기시킬 수 있어야 할 뿐만 아니라 특정 리비전에 해당되는 관련 파일(매뉴얼, 이론편서, GUI 정보, 비주얼 워크스페이스, 등...)을 동시에 분기해야 한다. 만약, 같은 리비전에 속하는 파일들이 존재하지 않는다면, 한 단계 낮은 리비전을 가진 파일들을 분기하여 분기된 관련파일들과 같이 새로운 정보로 등록된다. (그림 6)은 중간산출물이 최종결과물로 분기되어 저장되는 과정을 보여준다.

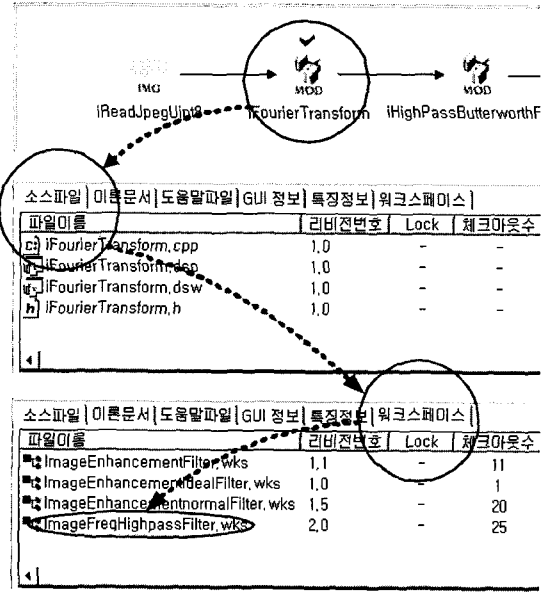
### 3. 구현 및 결과

제안된 시스템의 구현 환경은 다음과 같다.

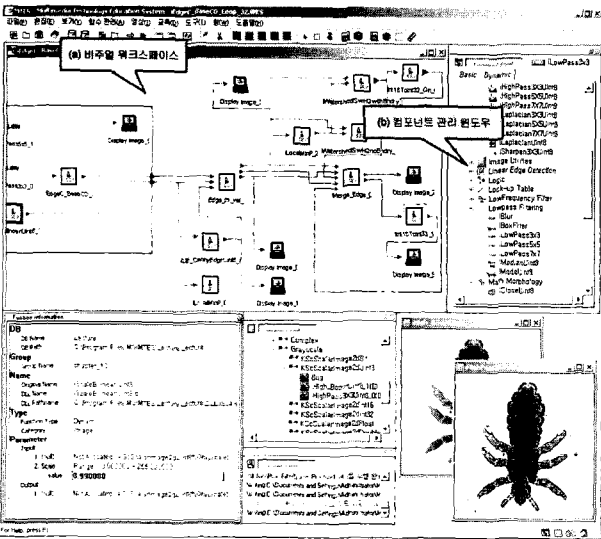
- 서버 사양
  - Windows 2000 Server / MS-SQL / TCP/IP
- 클라이언트 사양
  - Windows 2000 이상 / Pentium III 이상
- 개발언어 - Visual C++ (MFC) / ASP
- 영상처리 알고리즘 개발 환경
  - Khoros[20] / MTES[22]

(그림 7)은 본 논문의 실험에 사용된 영상처리 알고리즘 개발 환경의 메인 화면이다. (그림 7)의 영상처리 알고리즘 개발 환경은 영상처리 알고리즘을 아이콘 형태의 컴포넌트로 저장하여 비주얼 프로그래밍 환경을 이용한 시뮬레이션 및 응용시스템 개발이 가능한 환경을 제공한다. (그림 8)은 본 논문에서 제안한 시스템의 클라이언트 메인 화면으로 (그림 7)과 같은 유사한 관리 구조를 가지도록 하여 사용의 편리성을 주었다. (그림 8)의 좌측(a)에는 영상처리 컴포넌트들의 트리 구조가 나와 있으며, 유사한 기능의 컴포넌트

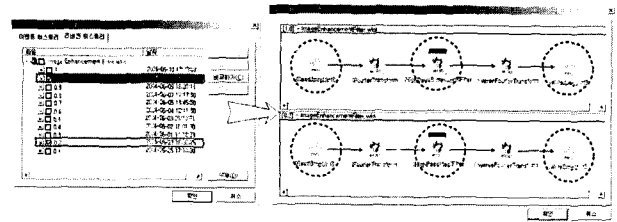
들이 묶여서 여러 개의 그룹으로 되어 있다. 좌측의 컴포넌트 트리에서 그룹을 선택하면 그룹에 속해있는 컴포넌트들이 나타나고, 그 컴포넌트 중 하나를 선택하게 되면, 우측에 컴포넌트와 관련된 정보들이 나타나게 된다. 우측상단(b)에는 컴포넌트를 사용하는 대표적인 비주얼 워크스페이스의 내용이 디스플레이 되며, 우측하단(c)에는 선택된 컴포넌트의 소스 파일을 포함한 관련정보들이 나타난다. 컴포넌트와 관련된 각 정보들은 우측하단의 위에 있는 탭 바를 통해서 볼 수 있으며, 우측 상단의 비주얼 워크스페이스 디스플레이 화면에 표시된 비주얼 워크스페이스 내용에서 다른 컴포넌트를 더블 클릭하게 되면 하이퍼링크 정보에 따라 선택된 컴포넌트 정보로 이동하게 된다. (그림 9)는 워크스페이스의 하이퍼링크 정보를 사용하여 관련 정보를 검색하는 예제 화면을 보여주고 있다. (그림 8)의 우측하단의 대화상자(d)는 선택된 소스코드 파일의 개발 이력을 볼 수 있는 리비전 정보를 보여준다.



(그림 9) 하이퍼링크 예제

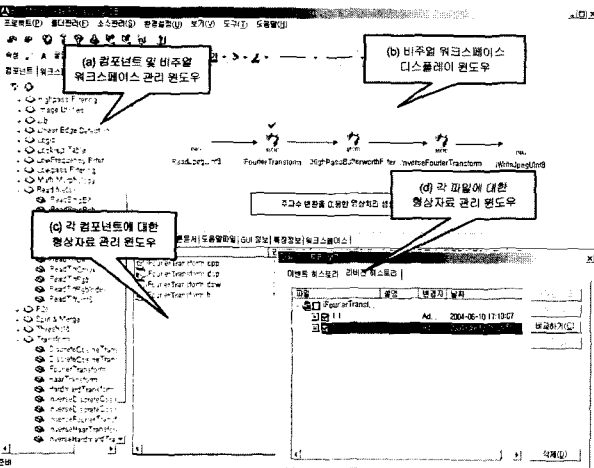


(그림 7) 비주얼 프로그래밍 환경 기반의 영상처리 알고리즘 개발 환경



(그림 10) 두 개의 비주얼 워크스페이스 파일간의 차이 비교하기

(그림 10)은 워크스페이스 파일의 다른 두 리비전 파일 사이의 차이점을 보여주는 기능이다. 0.2 버전과 0.7버전의 워크스페이스 파일에 달라진 부분을 보여주고 있으며, 달라진 부분은 아이콘의 위에 노란색의 바가 그려지게 되고, 같은 부분의 아이콘에는 아무 표시도 안된다. 빨간색 바가 그려진 아이콘 부분은 차이가 있는 아이콘 중에서 현재 마우스로 선택된 아이콘을 표시한다.



(그림 8) 제안된 SCM 시스템의 메인 화면

<표 1> 기존 시스템과의 특징 비교

- ① 제안 시스템    ② Clearcase[26]  
 ③ Harvest[27]    ④ PVCS[28]    ⑤ SourceSafe[29]

| 항 목                        | 시스템 | ① | ② | ③ | ④ | ⑤ |
|----------------------------|-----|---|---|---|---|---|
| 워크스페이스 기반 형상관리 체계          |     | ○ | × | × | × | × |
| 워크스페이스 가시화                 |     | ○ | × | × | × | × |
| 컴포넌트의 하이퍼링크 검색             |     | ○ | × | × | × | × |
| 워크스페이스와 인터페이스              |     | ○ | × | × | × | × |
| 중간산출물의 최종결과물로의 분기          |     | ○ | △ | △ | △ | △ |
| 컴포넌트 하이퍼링크 자동 생성           |     | ○ | × | × | × | × |
| 사용하고자 하는 알고리즘과 관련된 컴포넌트 검색 |     | ○ | × | × | × | × |

제안된 시스템과 같은 목적으로 개발된 시스템이 없어 정확한 비교가 어렵지만 기존에 알려져 있는 시스템들의 기능을 조사, 비교함으로써 제안 시스템의 특징을 제시하였다.

본 논문에서 제안한 시스템은 일반적인 SCM 시스템과 같은 텍스트 소스 코드 기반의 SCM 체계 및 기능을 동일하게 지원하고 있으며, 키워드 입력을 통한 검색기능도 동일하게 지원한다. 그러나 영상처리 알고리즘 개발 시에는 동일한 기능의 컴포넌트들이 다수 존재하는 영상처리 분야의 특징으로 인하여 기존 SCM 시스템들처럼 키워드 검색 기능을 통한 컴포넌트 이름만으로는 소프트웨어 개발에 적합한 컴포넌트를 찾기 어렵다. 뿐만 아니라 같은 컴포넌트라도 컴포넌트가 놓이는 위치에 따라서 다른 용도로 사용되는 경우도 많아 사용자들에게 정확한 사용을 유도해 주기 위한 비주얼 워크스페이스와 같은 사용 예를 제시방법이 필요하다. 그러나 기존 SCM 시스템들은 비주얼 워크스페이스의 디스플레이가 불가능하고 선택한 컴포넌트를 사용하기 위해서 필요한 컴포넌트가 어떤 것들이 있는지 메뉴얼이나 도움말을 모두 읽어야 사용이 가능하고, 해당 컴포넌트를 다운로드 받아보아야만 선택한 컴포넌트를 사용하기 위해 필요한 컴포넌트들이 무엇인지 알 수 있어 선택한 컴포넌트가 적합한 것인지, 사용에 필요한 다른 컴포넌트들이 있는지 확인하는데 불필요한 시간이 소요된다. 만일 기존 SCM 시스템에 제안된 형상관리 구조를 직접 입력하여 등록한다고 해도 제안된 시스템과 같이 하이퍼링크가 자동으로 생성되지 않게 되면 새로 등록되는 컴포넌트들과의 관계는 설정될 수가 없다. 즉, 제안된 시스템은 비주얼 워크스페이스 등록 시마다 컴포넌트 사이의 관계가 재설정됨으로써 기존 SCM 시스템에서 찾지 못했던 관계 들을 찾는 것이 가능하다는 장점을 가진다. 그리고 기본적으로 하이퍼링크에 사용되는 정보는 외부에 오픈되어 있는 정보가 아니라 SCM 시스템 내부에 존재하는 DB의 인덱스 정보이기 때문에 기존 SCM 시스템을 사용할 때 외부에서 하이퍼링크 정보를 생성하는 것이 불가능하다

제안 시스템은 영상처리 알고리즘 개발환경에 대해서 기존의 SCM 시스템에 비해 영상처리 알고리즘 개발환경에 특징들을 잘 지원해줄 수 있도록 개발되었다. 뿐만 아니라 비주얼 프로그래밍 환경을 기반으로 하는 시뮬레이션 환경을 체계적으로 지원해주는 비주얼한 관리 방법은 기존의 시스템들의 관리 방법에 비해 사용자들로 하여금 개발된 소프트웨어를 보다 쉽게 이해하고 사용할 수 있도록 유도해 줄 수 있어 재사용성을 극대화할 수 있다는 장점을 가진다.

#### 4. 결론 및 향후 연구방향

본 논문에서는 빠르게 성장하고 있는 영상처리 분야의 개발과 유지보수에 적합한 SCM 시스템을 제안하였다. 제안 시스템은 영상처리 개발 환경의 특징을 지원하고 문제점을 해결하기 위하여 개발되었으며, 실제적인 구현을 통하

여 영상처리 개발 환경에 적합한 SCM 시스템임을 확인할 수 있었다.

무엇보다도 일반적인 SCM 시스템과는 다른 비주얼 프로그래밍 도구를 기반으로 한 시뮬레이션 환경의 비주얼 워크스페이스와의 연계를 통하여 알고리즘의 사용 및 다른 알고리즘과의 관계에 대한 가시성을 극대화 하였으며, 이를 통한 소프트웨어에 대한 이해력을 한층 증가시켜 컴포넌트의 사용자, 개발자, 관리자로 하여금 보다 체계적인 관리 및 재사용성을 극대화할 수 있었다.

제안 시스템의 비주얼 워크스페이스에 기반한 형상관리 기능을 일반 업무절차용 소프트웨어 개발용으로도 확장할 수 있다면 일반적인 SCM 시스템보다는 가시성 및 재사용성을 개선할 수 있으리라 여겨진다.

#### 참 고 문 헌

- [1] Jacky Estublier, *Software Configuration Management : A Roadmap*, ICSE - Future of SE Track, 2000.
- [2] Reidar Conradi, *SPI frameworks : TQM, CMM, SPICE, ISO 9001, QIP Experiences and trends Norwegian SPIQ project*.
- [3] Magnus Larsson, *Applying Configuration Management Techniques to Component-Based Systems*, MRTC Report 00/24, 2000.
- [4] Axel Mahler, Andreas Lampen, *A toolkit for software configuration management*, 1988.
- [5] R. Conradi and B. Westfechtel, *Configuring Versioned Software Product*, SCM-6 Workshop. pp.88-109. Springer LNCS 1167. Berlin, March 1996.
- [6] S. Dart., *Concepts in Configuration Management Systems*, Proc. of the 3rd. Intl. Workshop on Software Configuration Management. Trondheim, Norway, June, 1991.
- [7] J. Estublier, *Workspace Management in Software Engineering Environments*, SCM-6 Workshop. Springer LNCS, 1167. Berlin, Germany, March 1996.
- [8] J. Estublier and S. Dami and M. Amieur, *High Level Process Modeling for SCM Systems*, SCM 7, LNCS 1235. pages 81~98, May, Boston, USA, 1997.
- [9] P. Feiler, *Configuration management models in commercial environments*, Technical report CMU/ SEI-91-TR-7. SEI 1991.
- [10] D. B. Leblang, *The CM Challenge : Configuration Management that Works*, Configuration Management, Edited by W. Tichy; J. Wiley and Sons. Trends in software, 1994.
- [11] Walter F. Tichy, *Tools for software configuration management*, In Proc. of the Int. Workshop on Software Version and Configuration Control, pp.1.20,

Grassau, January. 1988.

[12] R.C. Water, Automated software management based on structural models, Software Practice and Experience, 1989.

[13] Peter Herzum, Oliver Sims, Business Component Factory : A Comprehensive Overview of CBD for the Enterprise, OMG press, December, 1999.

[14] Darts, S., Concepts in Configuration Management Systems, Proceedings of Third International Conference on SCM, Trondheim, Norway, June, 12-14, pp.18, 1991.

[15] 양해술, 이하용, 소프트웨어 통합 환경구축을 위한 도구의 분석과 설계 방안, 한국정보처리학회, pp.834~845, 1996년 7월.

[16] William K. Pratt, Digital Image Processing : Luminance edge detector performance, A Wiley -Interscience publication, John Wiley & Sons, inc., pp.532~547, 1991.

[17] P.K.Sahoo, S.Soltani, A.K.C.Wong, and Y.C.Chen, A Survey of Thresholding Technique, Computer Vision, Graphics and Image Processing 41, pp.233~260, 1988.

[18] I.Pitas and A.N. Venetsannopoulos, Nonlinear Digital Filter : Principles and Applications, Kluwer Academic Publishers, pp.302~306, 1990.

[19] 이정헌, 채옥삼, 컴퓨터비전과 영상처리 알고리즘의 유지관리와 재사용을 위한 통합개발환경, 정보과학회 논문지 B, 제24권 제3호, 1997.

[20] JeongHun Lee, Oksam Chae, Multimedia data processing algorithm development environment MADE), Visual Data Exploration and Analysis VI, Proc. Of SPIE, pp.193~203, 1999.

[21] Khoros, <http://www.khoral.com>

[22] Wits, <http://www.logicalvision.com>

[23] Hello-Vision, <http://www.mtes.co.kr>

[24] DirectX graphedit, <http://msdn.microsoft.com>

[25] Matlab, <http://www.mathworks.com>

[26] ClearCase, <http://www-306.ibm.com/software>

[27] CCC/Harvest, <http://www3.ca.com>

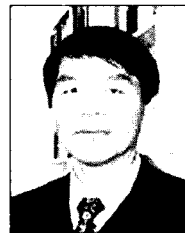
[28] PVCS, <http://www.merant.com>

[29] SourceSafe, <http://msdn.microsoft.com>



**이정헌**

e-mail : [opendori@paran.com](mailto:opendori@paran.com)  
 1992년 경희대학교 전자계산공학과(학사).  
 1994년 경희대학교 전자계산공학과  
 (공학석사).  
 1999년~2004년 MG Systems 부설 연구소  
 연구소장  
 2005년 경희대학교 컴퓨터공학과 (공학박사)  
 2004년~현재 자르테크 개발이사  
 관심분야 : 영상처리, 컴퓨터비전, 미디어센터, 소프트웨어 공학



**채옥삼**

e-mail : [oschae@khu.ac.kr](mailto:oschae@khu.ac.kr)  
 1977년 인하대학교 전자공학과(학사)  
 1982년 오클라호마 주립대학 전기 및  
 컴퓨터공학과(공학석사)  
 1986년 오클라호마 주립대학 전기 및  
 컴퓨터공학과(공학박사)  
 1986년~1988년 Texas Instrument Image Processing Lab. 선  
 임연구원  
 1988년~현재 경희대학교 전자계산공학과 교수  
 관심분야 : 컴퓨터비전, 컴퓨터 그래픽스