

< 논문 >

모션 텍스처를 이용한 차량 시뮬레이터의 통합

차 무 현[†] · 한 순 흥^{*}

(2006년 12월 28일 접수, 2007년 6월 22일 심사완료)

A Data Driven Motion Generation for Driving Simulators Using Motion Texture

Moohyun Cha and Soonhung Han

Key Words : VR(가상현실), Driving Simulator(차량시뮬레이터), Motion Texture(모션 텍스처)

Abstract

To improve the reality of motion simulator, the method of data-driven motion generation has been introduced to simply record and replay the motion of real vehicles. We can achieve high quality of reality from real samples, but it has no interactions between users and simulations. However, in character animation, user controllable motions are generated by the database made up of motion capture signals and appropriate control algorithms. In this study, as a tool for the interactive data-driven driving simulator, we proposed a new motion generation method. We sample the motion data from a real vehicle, transform the data into the appropriate data structure(motion block), and store a series of them into a database. While simulation, our system searches and synthesizes optimal motion blocks from database and generates motion stream reflecting current simulation conditions and parameterized user demands. We demonstrate the value of the proposed method through experiments with the integrated motion platform system.

1. 서 론

차량 시뮬레이터의 운동감을 생성하는 과정은, 일반적으로 차량의 동역학 모델링 및 운동방정식의 실시간 계산 과정을 통해 구현된다.^(4,9,10) 최근에는 이러한 전통적인 방법 외에도 현실감을 증대시키거나 기존의 시스템에 병행 적용될 수 있는, 다양하고 새로운 운동감 생성법이 연구되고 있다.^(11,12) 특히, 동역학 모델링 방식에 의해 계산하기 힘들거나 실시간 시뮬레이션이 불가능한 복잡한 실제계 운동감 재현을 위해, 실제 탑승체의 주행 운동감을 획득하고, 이를 시뮬레이터에 직접 적용하는 데이터 기반 시스템이 개발된 바 있다.⁽¹⁴⁾ 이 방법은 실제 운동감을 제공하므로 현실

감의 확보가 용이하나, 현재로는 사용자와의 상호작용이 없이 가상 콘텐츠만을 단순히 제공하는 단계에 머물고 있다.

한편, 컴퓨터 그래픽스의 캐릭터 애니메이션 분야의 경우, 약 10년 전부터 자연스러운 인간의 동작을 모사하기 위해 동작 캡처 기술(Motion Capture, Mocap)이 널리 활용되어 왔으며, 최근에는 이를 아바타에 그대로 적용하는 단계를 뛰어 넘어, 캡처된 데이터베이스를 활용한 동작 합성 및 실시간 제어에 관한 연구가 수행되고 있다.^(3,6,7)

본 연구에서는 위와 같은 동작 캡처 데이터 기반의 합성 및 제어 기술들을 차량 시뮬레이터의 운동감 생성에 적용하였다. 이 과정에서 실제 운동감을 그대로 활용할 수 있는 장점을 유지하면서, 운전자의 제어 요구에 따라 시뮬레이션 상황에 적절한 운동감의 생성이 가능한 새로운 운동감 생성법을 제안하며, 이를 능동형(Active) 데이터 기반 운동감 생성법이라 정의한다.

이는 Fig. 1 과 같이 실제 차량의 주행 정보를 획득

[†] 책임저자, 회원, 한국과학기술원 기계공학과,
(현재) 한국기계연구원
E-mail : mhcha@kimm.re.kr
TEL : (042)868-7927 FAX : (042)868-7418
^{*} 회원, 한국과학기술원 기계공학과

하여 운동감 데이터베이스를 구축하는 과정(Building Motion Database)과 구축된 데이터베이스로부터 적절한 운동감을 선택 및 합성하여 실시간 운동감을 생성하는 과정(Interactive Motion Generation)으로 나뉜다.

본 연구에서 제안하고자 하는 능동형 데이터 기반 운동감 생성 시스템은 높은 현실감의 확보, 세부 운동감의 합성, 기존 시스템과의 병행성 및 차량 고유 특성에 따른 변경이 용이하다는 장점을 가지며, 이를 적용한 실제 차량 시뮬레이터 시스템의 구현 및 통합과정을 통해 그 유용성 및 응용방안에 대해 소개하고자 한다.

2. 관련 연구

2.1 데이터 기반 접근법

데이터 기반 접근법(Data-Driven Approach)은 실세계에 깊이 내재된 원리(Principal)를 이해하기에는 너무 복잡하기 때문에 모델링 대신 실세계 현상을 샘플링하고 이를 직접 사용하고자 하는 아이디어에서 출발하였다. 카메라에 의한 실사(Picture), 캠코더에 의한 영상(Video), 모션 캡처에 의한 동작(Motion), 3D 스캐너를 통한 형상(Geometry), 녹음기에 의한 음성, 소리(Voice/Sound), 기타 물리적인 특성들이 이런 데이터로서 활용 될 수 있다. 이 접근법의 목적은 실세계를 계산하여 구축하는 것이 아니라 컴퓨터에 저장된 수많은 예제(Sample)로부터 실세계를 재구성(Reconstruct)하는 것이며, 재구성을 통해 실세계에 존재하지 않는 가상환경을 만들어 내기(Create)도 한다.⁽¹³⁾

2.2 수동형(Passive) 데이터 기반 운동감 생성

실측한 데이터에 기반한 운동감 생성 방법은 차량에 직접 센서를 탑재하고 주행 중 측정된 운동 데이터를 활용하여 운동감을 생성하는 방식이다. ㈜에이알비전은 롤러코스터, 자동차, 경비행기, 급류 보트 등의 탑승구에 3축 가속도 센서와 각속도 센서 및 비디오 캠코더를 부착한 뒤, 주행 운동 정보를 획득하고 영상을 촬영하여 이를 그대로 재생하는 수동형(Passive) 데이터 기반 방식의 차량 시뮬레이터를 개발하였다.⁽¹⁴⁾ 이 방식은 물리법칙에 의해 계산된 운동감이 아닌 실제 측정 데이터를 사용하기 때문에 현실감 높은 운동감을 제공할 수 있지만, 사용자와의 상호작용이 배제된 단순한 콘텐츠의 제공만이 가능하다는 단점을 가진다. 따라서 현실성 높은 데이터를 그대로 활용할 수 있는 장점을 살리면서 운전자의 제어 요구에 따라 적절한 운동감을 생성할 수 있는 능동형

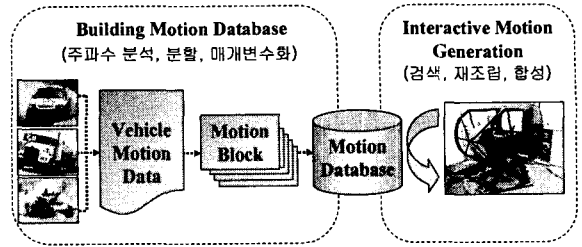
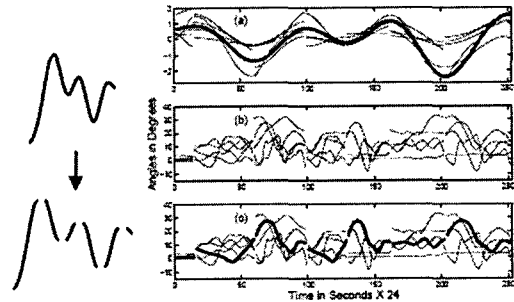


Fig. 1 Data driven motion generation



(1) Segmentation (2) Path finding & texturing

Fig. 2 Motion texturing procedure⁽⁸⁾

데이터 기반의 운동감 생성법이 필요하다.

2.3 데이터 기반 캐릭터 애니메이션

데이터 기반 캐릭터 애니메이션(Data-Driven Character Animation) 기술은 실세계 예제에서 캡처된 동작 데이터를 사용하는 기술로서, 사람(실험자)의 여러 관절 부위에 부착된 마커의 위치를 측정하여 데이터베이스에 저장한 다음, 실시간 애니메이션 생성 시, 이를 합성 및 제어하여 원하는 애니메이션을 생성해 내는 방법이다. 이 방식은 도메인 지식이 불필요 하며 미묘한 동작 부분까지 데이터로 표현될 수 있기 때문에, 현재 상용 애니메이션 시스템에 널리 사용되고 있으며, 그 하위 기술로는 전문 편집가의 수작업을 통한 애니메이션 저작, 동작의 단순한 아바타 매핑을 수행하는 Puppetry 기술, 데이터로부터 일반화된 모델을 유추하는 Generalized Motion 기술, 마지막으로 데이터를 적절하게 분할하고 이를 효과적인 순서로 재배열하여 데이터 획득시의 의도와 정보를 가능한 보존할 수 있는 분할 및 재조립(Cutting & Reassemble) 기술 등이 있다.

분할 및 재조립 기술 중 모션 텍스처(Motion Texture)라 불리는 기술은 Fig. 2와 같이 다양한 주파수 범위의 동작 캡처 데이터를 2D 이미지 텍스처 매핑처럼 활용하고 이를 기존의 동작에 단계적

으로 적용하는 방식으로서, 동작 데이터를 적절히 분할하고[Fig. 1-(1)] 저주파 신호(blue)에 매칭되는 고주파 신호(red)를 검색 및 합성하는 과정을 통해 전체적인 동작을 생성하게 된다.⁽⁸⁾

본 연구에서는 위와 같이 캐릭터 애니메이션 분야에서 활발하게 연구되고 있는 데이터 기반 동작 생성 기술, 특히 모션 텍스처 합성/제어 기술을 활용하여 제어가 가능한 데이터 기반 시뮬레이터 운동감 생성법을 제안한다. Table 1에서는 인간의 일부 동작을 입력 받아 전체 동작을 생성하는 모션 텍스처 기술과 본 연구의 운전 명령 입력에 대한 차량 동적 거동을 생성하는 과정에 사용된 세부 기술들을 서로 비교하였으며 자세한 사항은 다음 절에 소개하였다.

3. 데이터 기반 운동감 생성

본 연구에서 제안하고자 하는 능동형 데이터 기반 운동감 생성의 전체 과정을 Fig. 3에 도식화 하였다. 먼저 오프라인 절차로서, 차량의 주행정보를 획득하여 운동감 데이터베이스를 구축하는 과정이 있고, 실시간 시뮬레이션 시에 미리 구축된 운동감 데이터베이스로부터, 운전자와 상호작용하며 운동감을 생성하는 온라인 과정(Motion Blending)으로 구성된다.

운동감 데이터베이스를 구축하는 과정에서는 실제 운항중인 탑승체(자동차)로부터 운전 조작 명령 및 6 자유도 동적 운동량을 측정하여 특정 차량에 대한 주행 정보(Motion Samples)를 수집한다. 이 데이터는 노이즈 제거 및 서로 다른 동적 특성 분리를 위해 주파수대에 따른 분석(Frequency Analysis)과 특징 운동감 추출 과정을 거치게 되며, 이러한 일련의 데이터 집합을 시간축에 따라 적절히 분할(Segmentation)하여 다수의 운동감 조각(Motion Block)을 생성하게 된다. 이 운동감 조각에는 동적 운동량뿐 아니라 함께 측정된 운전 명령 신호가 저장되어 있으며, 실시간 검색의 효율성을 위해 매개변수화(Parameterization) 및 전처리 계산 과정을 거쳐 데이터베이스에 저장된다.

실시간 시뮬레이터 구동 시에는 앞서 구축된 모션 데이터베이스에서 현재 운전조작 및 시뮬레이션 상태에 해당하는 최적의 운동감 조각을 검색/선택(Search/Select)하고, 필요 시에는 운동감 근사 과정(Approximation)을 수행한다. 이렇게 얻어진 운동감 조각은 시간축 또는 공간축에 대해 합성(Synthesis/Texturing)되어 최종적인 운동감 스트림을 생성하게 되며 이는 모션 베이스(운동판)에 전

Table 1 Comparison of motion texturing⁽⁸⁾ and this research

	모션 텍스처	본 연구
대상	캐릭터 애니메이션	운동 시뮬레이터
데이터 베이스	인간 동작 캡처	차량 주행 데이터
입력	일부 동작	운전 명령
출력	전체 동작	차량 운동감
모션 추출	주파수 분석	주파수 분석
분할 방식	실시간 판단	동일 간격
매칭 방식	입력값 유사성 DB 연결성	입력값 유사성 입력값 연속성 입력값 우량성 DB 연결성
보간 방식	Time Warping	Curve Fitting
합성 방식	Motion Warping	Motion Warping

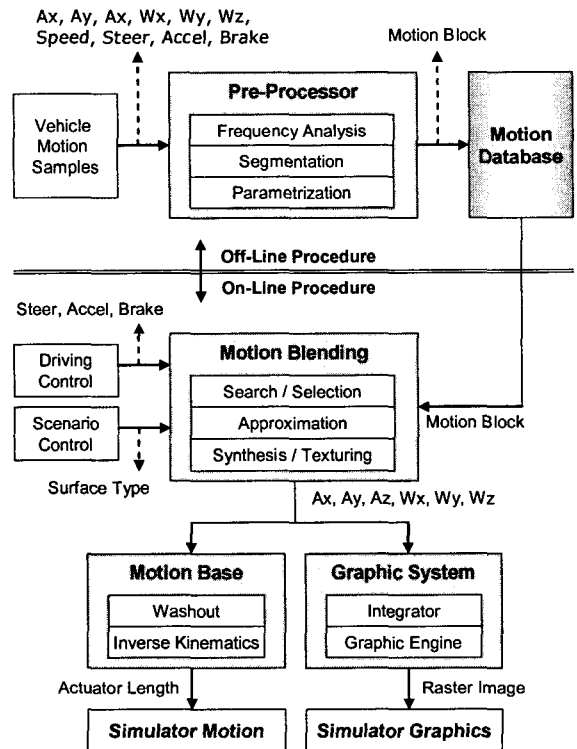


Fig. 3 Data Driven Motion Generation

달되어 액추에이터를 구동하거나 그래픽 시스템에 전달되어 이미지를 생성하게 된다.

이러한 능동형 데이터 기반 운동감 생성시스템에서의 핵심적인 부분은 운동감 데이터베이스를 어떻게 효과적으로 구축할 것인가와 실시간 시뮬

레이션 시 어떤 방법으로 최적의 운동감 조각들을 찾아내어 합성할 것인가이다.

3.1 차량 주행 데이터의 획득

획득하고자 하는 차량 주행 데이터 중 가장 중요한 것은 (1) 차량 강체의 동적 운동량을 나타내는 6 자유도 운동 데이터(Driving Motion), (2) 현재 운동량을 표현하는 속도, (3) 차량의 거동을 제어하는 운전 제어 데이터(Driving Control)이다. 또한, 이러한 차량의 거동은 외부 주행 환경(도로표면의 종류, 경사도, 충돌, 공기저항 등)에 의해서도 영향을 받을 수 있으므로, 외부 운전 환경(Driving Environment)에 대한 데이터도 획득하여야 하지만, 본 연구에서는 도로 표면의 종류 정보만을 획득하여 적용하고자 하며, 이외의 환경 변수들은 추후 연구로 남겨 두고자 한다. Table 2 는 이러한 차량 주행 데이터의 구조를 정의하고 있다.

3.2 주파수 분석 및 특징 운동감 추출

차량 주행 데이터는 다양한 운전 제어에 의해 생성된 운동 성분, 도로 표면 등의 운전 환경에 의한 성분, 차량 고유 진동 성분 등이 합성된 복합적인 신호들로 구성되어 있다. 따라서 시뮬레이터의 다양하고 효율적인 제어를 위해 이를 차량 거동 원인을 기준으로 분류하여 실제 의미 있는 특징 운동감을 추출하는 과정이 필요하다.

르노 차량 시뮬레이터(Renault Driving Simulator) 연구에서는 실차 주행 테스트를 통해 얻은 차량의 가속도를 주파수대별 분포도로 나타내어, 인간이 감각적으로 느낄 수 있는 주파수 영역인 0~15Hz 대에 대한 그 거동의 원인을 분석하였다.⁶⁾ 0~1Hz 구간은 운전자의 운전 명령(가속, 조향각)에 의한 차량의 거동 성분, 1Hz 이상 구간은 도로 표면 상태 등이 반영된 현가장치, 차체, 난류 등에 의한 진동 성분으로 해석하였다. 본 연구에서도 주파수 분석을 실행하였으며, 운전 조작에 의한 차량의 거동(저주파 영역, 0~1Hz)과 노면 상황 또는 차체 고유 진동에 의한 거동(고주파 영역, 1~15Hz) 및 시뮬레이터에 의해 표현 불가능한 노이즈 영역(15Hz~)으로 분리할 수 있었다.

3.3 운동감 조각

주파수 분석 과정을 거친 연속적인 주행 데이터들은 최소한의 연속성을 유지하고, 실시간 시뮬레이션 성능을 만족하며, 운전 제어에 따른 선택 및 합성을 위해, 여러 개의 단위(프레임)로 집산화하여 분할될 필요가 있다. 본 연구에서는 이 그룹의 단위를 운동감 조각(Motion Block)이라 정의하였으

Table 2 Data structure for vehicle driving

Category	Field	Description
Driving Motion (차량 거동)	Ax (Surge)	병진 가속도(x)
	Ay (Sway)	병진 가속도(y)
	Az (Heave)	병진 가속도(z)
	Wx (Roll)	회전 각속도(x)
	Wy (Pitch)	회전 각속도(y)
	Wz (Yaw)	회전 각속도(z)
Driving Control (운전 제어)	Speed	차량의 속력
	Accelerator	가속 페달의 깊이
	Brake	감속 페달의 깊이
Driving Environment (운전 환경)	Steering	조향 각도
	Event1	도로 표면의 종류
	Event2	굴곡(경사), 충돌 등

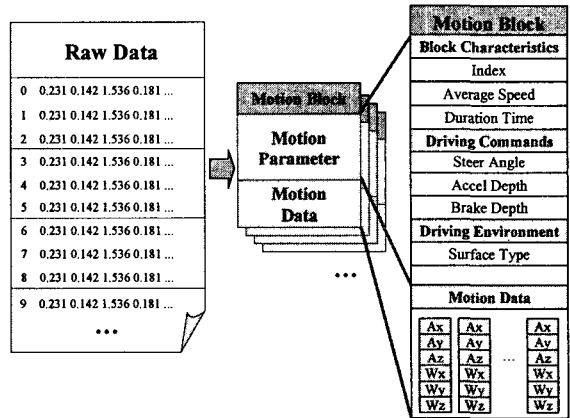


Fig. 4 Motion block & data structure

며, Fig. 4 에서는 운동감 조각의 데이터 구조를 보여주고 있다. 운동감 조각은 실시간 비교 검색을 위한 매개변수 영역과 실제 차량 운동 데이터를 저장하는 운동감 데이터 영역으로 구성된다.

매개변수 영역은 운동감 조각의 고유 아이디, 평균 속력, 지속 시간 등을 저장하는 특성 영역(Block Characteristics)과 조향각, 가속기, 감속기의 시각값, 종료값, 변화를 등을 저장하는 운전제어 영역(Driving Commands) 그리고 도로 표면 특성을 나타내는 운전 환경 (Driving Environments) 영역으로 구성된다. 또한, 운동감 데이터 영역은 단위 지속시간 동안 샘플링된 6 자유도 운동 데이터의 집합으로 구성된다.

단위 조각의 분할 기준에 있어, 모션 텍스처 방식에서는 신호 변화율의 부호가 바뀌는 지점을 사용하여 실시간 곡선 비교 및 Time Warping⁽¹⁾ 과정을 수행하였으나, 본 연구에서는 실시간 시뮬레이션 성능을 만족하기 위해 운전 제어와 깊은 연관성을 가지는 저주파 운동감은 0.2초 단위, 제어값과 큰 관련이 없는 고주파 배경 진동의 경우 2 초 구간으로 분할하였으며, 각 운전 제어 신호 구간에 대한 전체적인 스펙트럼의 모양보다는 구간의 시작값 및 종료값으로써 해당 구간의 제어 특성을 정의하였다.

3.4 운전 조건 매개변수의 획득

운동감 데이터베이스로부터 실시간 운동감 스트림을 생성하기 위한 첫 번째 단계는, 데이터베이스에 저장된 운동감 조각 매개변수와 비교하게 될 동일한 시간 간격의 실시간 운전 조건 매개변수를 획득하는 것이다. 이를 위해 조향각, 가속속기의 값이 등에 대한 시뮬레이터 운전 조작 명령을 주기적으로 수집하고, 시뮬레이션 과정에서 계산된 차량의 속력 및 주행 환경 데이터를 획득하여, 각 데이터베이스 내의 매개변수와 동일한 형식의 구조를 가지는 연속된 데이터를 생성하였다.

3.5 최적 운동감 조각의 검색과 선택

온라인 과정의 두 번째 단계인 최적 운동감의 선택은 실시간 성능 확보를 위해 매개변수와 가장 유사한(Similar) 운동감 조각 후보를 검색하는 과정과, 이들 후보 중 가장 적절한 운동감 조각을 선택하는 과정으로 구성하였다. 이때, 시스템의 성격상 바로 앞에서 주어진 운전 조작 명령의 변화 상태에 의해서도 현재의 거동 특성이 달라질 수 있기 때문에, 바로 직전 조각과의 매개변수 연속성(Continuity)이 고려되어야 한다. 또한, 샘플링된 데이터의 집합은 샘플링 순서대로 재생하는 것이 가장 자연스럽게 때문에 직전 조각과의 연결성(Connectivity)도 고려한다. 마지막으로, 운동감 조각 분할에서 기본 가정된 매개변수의 선형성을 보장하기 위해, 비선형 매개변수를 포함하는 운동감 조각은 그 우량성(Quality)이 낮다고 판단하여 우선 순위에서 낮게 책정하였다. 본 연구에서는 이들을 모두 반영하여 실시간 계산이 가능한 선택비용 평가함수(Selection Cost Evaluation Function)를 식 (1)과 같이 정의하여 사용하였으며, Table 3에서는 전체적인 최적 운동감의 선택 알고리즘에 대해 설명하고 있다.

각 항(term)은 매개변수의 유사성, 연속성, 연결성, 우량성 등을 평가하며, 각 항의 우선순위는 운

Table 3 Algorithms for selecting the best motion block

Algorithms for Selecting the Best Motion Block	
1. Obtaining Driving Commands	- 실시간 운전 조건 매개변수 획득
2. Searching for Similar Candidates	- 매개변수와 흡사한 운동감 조각 후보 검색
3. Evaluating Selection Cost	- 선택 비용 평가 수행
4. Choosing the Best Motion Block	- 최소 선택비용 운동감 조각 선택

$$Fsc(i, j) = \alpha d(P_i, P_j) + \beta d(D_i, D_j) + \gamma C(j, j-1) + \delta Q(j) \quad (1)$$

Fsc : Selection cost function
i : Simulation parameter id *j* : Database parameter id
P : Motion block parameter **D** : Parameter derivative
C : Connectivity (connected: 0, not connected: 1)
Q : Quality (nonlinear count)
d(a, b) : Difference between a and b
 $\alpha, \beta, \gamma, \delta$: Coefficient for priority

$$s(t) = \frac{d(t)}{2} \left(\cos \frac{\pi t}{N-1} + 1 \right), \quad t = 0, 1, 2, \dots, N-1 \quad (2)$$

동감 데이터베이스의 구축과정 및 실시간 시스템의 특성에 따라 달라질 수 있기 때문에 우선순위 계수를 두어 평가하였다. 식 (1)에서의 비교 대상은 실시간 운전 조건 매개변수(*i*)와 데이터베이스에 저장된 운동감 조각의 매개변수(*j*)이다. 두 매개변수는 동일한 데이터 구조를 가지므로 쉽게 평가가 가능하다. 유사성은 각 매개변수 값(**P**)의 차이로써 비교하며, 연속성은 각 매개변수의 변화율 값(**D**)의 차이로써 비교한다. 또한, 현재 비교되고 있는 데이터베이스 내의 운동감 조각(*j*)과 바로 직전 과정에서 선택된 운동감 조각(*j-1*)과의 고유 아이디를 비교함으로써, 그 차이가 +1 이 아니면 원래 샘플링 과정 중 연결된 데이터라 할 수 없으므로, 이를 반영하는 항(**C**)을 두었다. 마지막으로, 현재 비교되는 운동감 조각(*j*)에 저장된 매개변수의 비선형 데이터 개수를 나타내는 항(**Q**)을 두어 평가에 반영하였다.

3.6 운동감 조각 근사

운전 조건 매개변수를 통한 최적의 운동감 검색 및 선택 알고리즘은 데이터베이스에 원하는 운동

감 조각이 존재할 경우 가장 효과적이지만, 만약 주어진 한계 내에서 검색하지 못할 경우, 곡선 맞춤(Curve Fitting)에 의한 근사 및 최소 자승법(Least Square Method)에 의한 보간을 적용하거나, 검색 조건 한계(threshold)를 완화하여 다시 검색하는 방법을 사용하였다.⁽¹²⁾

3.7 운동감 합성

운동감의 합성은 3.6 절에서 생성된 새로운 운동감 조각을 시간 축을 기준으로 서로 연결하는 것(Synthesis)과 고주파 운동감과 저주파 운동감에 대한 공간상의 합성(Texturing)으로 나뉘어 진다. 공간 축 합성은 단순한 선형 중첩 합성(Linear Superposition)을 사용하였으며, 시간 축 합성은 Motion Warping⁽²⁾기술을 사용하여 부드러운 연결성을 가지는 운동감 스트림을 생성하였다. 식 (2)에서는 Motion Warping 기법에 합성함수로서 사용한 Scalar Blending Function 수식을 보여주고 있으며, $s(t)$ 와 $d(t)$ 는 각각 시간 t 에 대한 합성결과와 두 운동감의 차이값을, N 은 전체 합성구간의 시간길이 나타낸다.

4. 실험 및 결과

4.1 시스템의 구현

차량 주행 데이터를 샘플링하기 위해, Fig. 5와 같이 실험 장비를 구성하였다. 6 자유도 운동 데이터의 경우, 3축 가속도 센서와 3축 자이로 센서가 함께 내장된 DMU(Dynamics Measurement Unit) 장비를 사용하였으며, 운전 제어 데이터의 획득을 위해 케이블 타입 변위센서 3 개를 차량의 핸들, 액셀레이터 및 브레이크 페달에 설치하였다. 실차 주행 실험은 자동차 부품연구원(KATECH)의 주행 성능시험장⁽¹⁷⁾에서 수행하였으며, Fig. 6과 같은 데이터 수집 응용프로그램을 개발하여 아스팔트 직선 주행로 및 벨지안로, 자갈길로, 빨래판로 등과 같은 9 종류의 특수 시험로에 대한 실차 주행 데이터를 확보하였다. 데이터 로깅 프로그램은 각 센서 장비에서 제공하는 API 를 호출하는 방식으로 구현하였으며, 동일한 샘플링 주기 획득을 위해, 명령(Command)과 반응(Response) 메커니즘을 적용하였다. 전체적인 데이터 획득 시스템의 구조 및 데이터 흐름도를 Fig. 7에 나타내었다.

본 연구에서 제한한 분할과 합성 기반의 운동감 생성 시스템을 검증하기 위해 차량 시뮬레이터 시스템을 구성하였으며, Fig. 8 이 그 구성도를 보여준다. 그래픽 시스템은 오픈 소스 기반의 레이싱

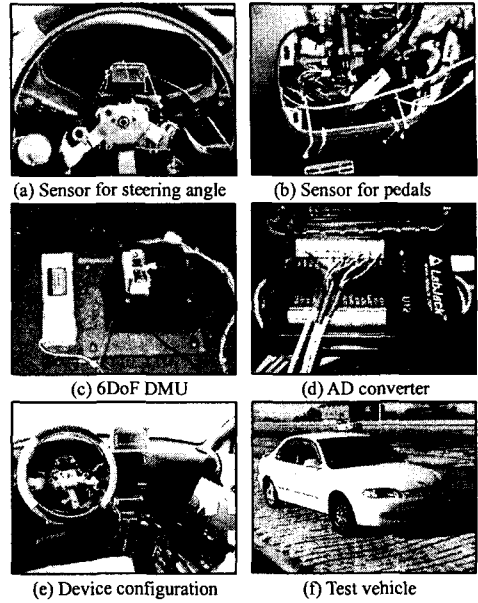


Fig. 5 Experimental devices & test environments

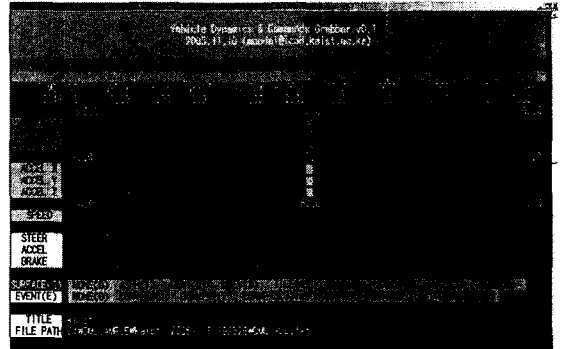


Fig. 6 Data acquisition SW

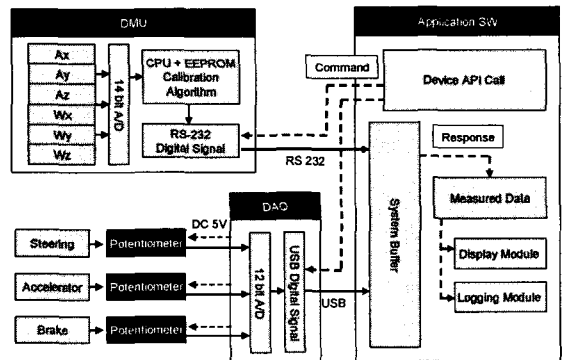


Fig. 7 Data flow diagram for data acquisition SW

게임 소프트웨어인 프랑스의 TORCS(The Open Racing Car Simulator, v.1.2.4)⁽¹⁶⁾를 사용하였고, 모션

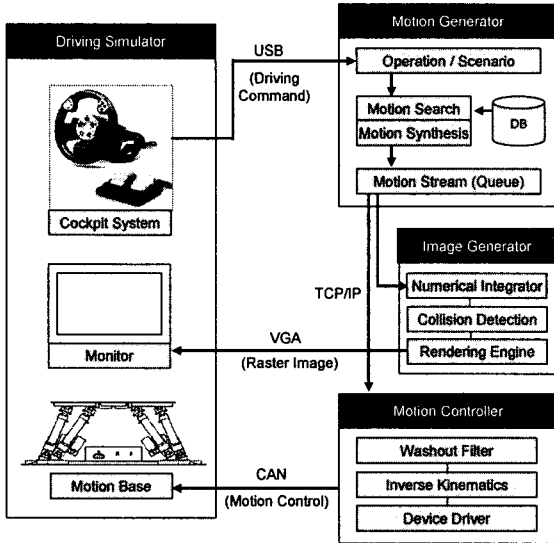


Fig. 8 System diagram of driving simulator

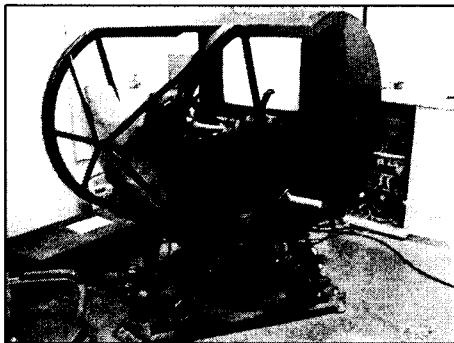


Fig. 9 Simuline DSMCP608

Table 4 Specification of driving simulator

명칭	Motion / Image Generator	Motion Controller
주요 기능	Motion Blending Image Generation Scenario Control User Input Database	Washout Filter Inverse Kinematics Hardware Control
설치 S/W	TORCS, MySQL, SMCP Client, Motion Generator	SMCP Server
CPU	P4 2.4 GHz	P3 800 MHz
RAM	512M DDR	128M SDRAM
HDD	120G IDE (7.2K)	20G IDE (5.4K)
OS	Windows 2000	Windows 98

베이스로는 Fig. 9 와 같은 (주)시플라인의 DSMCP608⁽¹⁵⁾ 기종을 적용하였으며, 제어용 S/W 로는 SMCP⁽¹⁵⁾ Server/Client 를 사용하였다. 그리고 데이터 기반

운동감 생성 모듈(Motion Generator)을 직접 개발하여 TORCS 에 삽입하였다.

전체 시스템은 운동감 생성기/그래픽 생성기/시나리오 제어기/데이터베이스를 포함하는 운동감 생성 서버와, 모션 베이스를 제어하기 위한 운동감 제어 서버, 총 2 개의 서버로 구성하였으며, 추후 시뮬레이션 성능을 위해 분산 시스템으로 구축될 필요가 있다. Table 4 는 본 연구의 차량 시뮬레이터 시스템에 적용된 각 HW, SW 의 주요 기능 및 사양을 보여준다.

4.2 데이터 기반 운동감 생성 시스템의 검증

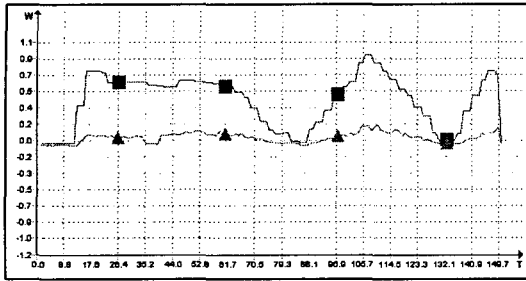
먼저 TORCS 에서 제공하는 동역학 모델 기반 운동감 생성 방식과 본 연구에서 제안한 데이터 기반 방식을 사용하여 차량 시뮬레이터를 각각 구동하고, 그 시뮬레이션 결과값을 그래프로 출력하였다. 고주파 운동감의 경우, 그 타당성 검증 방법이 다소 모호하기 때문에 비교하지 않았으며 (데이터 기반 방식이 월등하다), 운전 제어 입력(가속기, 조향각)에 대한 저주파 운동감 결과값(x, y 병진 가속도)을 서로 비교하였다. 실험 결과, Fig. 10 및 Fig. 11 과 같이 두 시스템이 서로 비슷한 곡선 추이를 나타내었으며, 가속도의 경우 실제 내연기관의 응답 지연현상 및 가속능력을 고려한다면, 가속기 입력과 거의 동일한 곡선 양상을 나타내는 동역학 모델 방식보다, 데이터 기반 방식이 좀 더 실제와 가깝다 판단할 수 있다. 하지만 조향각과 관련된 운동감의 경우, 샘플링 데이터의 부족 및 수치 적분 과정의 오차 누적으로 인해 그래픽 동시성에 있어 다소 부정확한 결과를 나타내었다.

4.3 하이브리드 운동감 생성

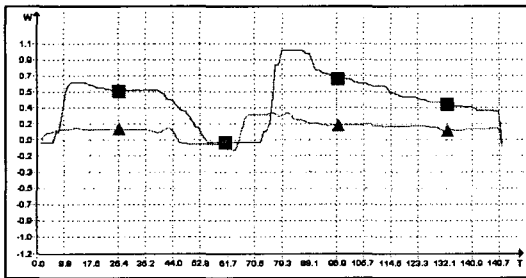
본 연구에서 제안하는 방식은 주파수 분석을 통해 저주파와 고주파 특징 운동감을 분리하였으므로 특정 목적을 위한 기존 시스템과의 병행이 용이하다. 그 예로서, 운전 조작 및 시나리오에 의한 저주파 운동감에 대해서는 기존 운동방정식 시스템을 활용하고, 차량 고유 진동 또는 노면 상황에 의한 배경 진동 등에 대해서는 데이터 기반 시스템을 적용하여 좀 더 현실적인 운동감을 제공할 수 있는 하이브리드 시뮬레이터 시스템을 구성할 수 있다. Fig. 12 는 하이브리드 시스템을 이용한 벨지안로 주행 모습을 나타내며, Fig. 13 에서는 기존의 운동감에 노면 특징 운동감이 합성된 결과를 보여주고 있다.

4.4 실시간 성능

성능 분석을 위해 운동감 생성 로직의 수행 시간을 측정하였다. DB 내의 운동감 조작 개수(일반

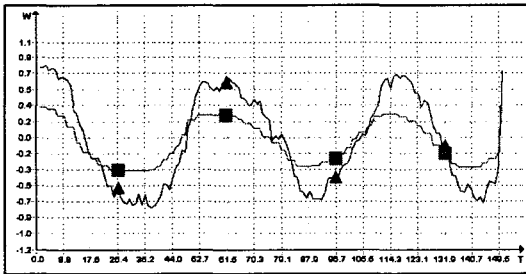


(a) Model driven

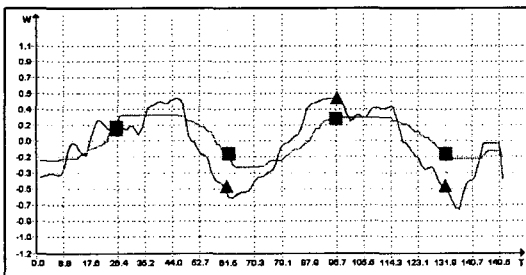


(b) Data driven

Fig. 10 Simulator motion graph
(■: Accelerator Depth, ▲: Linear Acceleration(x))



(a) Model driven



(b) Data driven

Fig. 11 Simulator motion graph
(■: Steering Angle, ▲: Linear Acceleration(y))

도로 4,679, 특수 도로 210)에 따라, 그 차이가 발생할 수 있겠지만, 멀티쓰레드 내에서 200 ms 단위로 발생되는 운동감 생성 요구에 대해 그 주기보

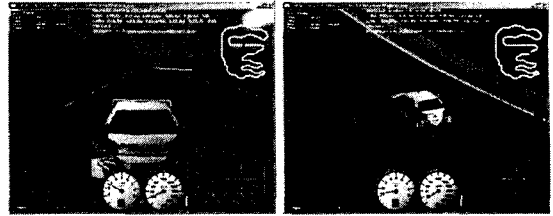


Fig. 12 Snapshot images (Belgian road condition)

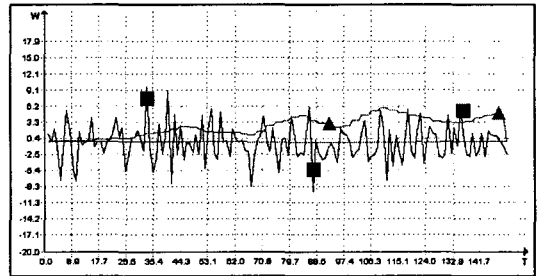


Fig. 13 Simulator motion graph (Belgian road)
(▲:Steering Angle, ■:Linear Acceleration(z))

다 훨씬 짧은 수행 시간(평균 17.9 ms)이 필요하였으며 결국, 실시간 성능을 확보했다 할 수 있다.

5. 결론

본 연구에서는 실측 데이터를 사용하여 제어 가능한 시뮬레이터 운동감을 생성할 수 있는 새로운 방법을 제안하였으며, 이는 운동감 데이터베이스를 활용하여 실시간으로 적절한 운동감을 검색, 선택, 합성하는 과정이라 할 수 있다.

이 방식은 측정된 현실 데이터를 사용하기 때문에 높은 현실감의 운동감 재현이 가능하며, 수학적 모델링이 힘들거나 불가능한 실제 또는 가상 대상체의 세부 운동감까지 재현이 가능하다. 또한 그 거동 원인별로 특징 운동감을 추출하였기 때문에 다양한 목적에 따라 기존 모델링 기반 시스템과의 병행이 가능하며, 운동감 데이터베이스는 정형화된 데이터구조로 이루어져 있으므로, 차량 실측 데이터의 변경만으로도 서로 다른 차량의 고유 진동 또는 동적 특성을 반영할 수 있다.

또한, 실시간 시뮬레이션이 불가능한 복잡한 공학적 모델에 대해 시나리오에 따른 해석을 미리 충분히 수행하고 이들을 매개변수화 하여 데이터베이스에 저장한 다음, 본 연구와 같은 적절한 선택 및 합성 알고리즘을 적용하여 인터랙티브 시뮬레이션 결과를 얻을 수 있으며, 결과적으로 해석

엔지니어링 데이터에 기반한 다양한 가상현실 시뮬레이터 구축이 가능하다.

본 연구에서 소개한 데이터 기반 방식의 기본적인 접근법은 세부 데이터의 보존이 가능한 분할 및 재조립 기법이며, 다소 단순한 방법에 의해 그 과정이 구현되었다. 캐릭터 애니메이션 분야의 경우, 다양하고 효율적인 알고리즘이 꾸준히 연구되고 있으며, 이를 적절히 활용하면 실시간 성능을 만족하고 공학적 데이터 활용에 최적화된, 개선된 데이터 기반 접근법 연구가 기대된다.

참고문헌

- (1) Bruderlin, A. and Williams, L., 1995, "Motion Signal Processing," *Proceedings of SIGGRAPH 95*, Vol. 29, pp. 97~104.
- (2) Witkin, A. and Popovic, Z., 1995, "Motion Warping," *Proceedings of SIGGRAPH 95*, Vol. 29, pp. 105~108.
- (3) Lee Jehee and Shin Sung Yong, 1999, "A Hierarchical Approach to Interactive Motion Editing for Human-like Characters," *Proceedings of SIGGRAPH 99*, pp. 39~48.
- (4) Lee, W-S., Kim, J.H., Cho, J.H. and Lee, S.J., 1999, "The Kookmin University Driving Simulators for Vehicle Control System Development and Human Factor Study," *Driving Simulation Conference '99*, pp. 75~86.
- (5) Reymond, G. and Kemeny, A., 2000, "Motion Cueing in the Renault Driving Simulator," *Vehicle System Dynamics*, Vol. 34, pp. 249~259.
- (6) Arikan, O. and Forsyth, D.A., 2002, "Interactive Motion Generation from Examples," *Proceedings of SIGGRAPH 2002*, Vol. 21, No. 3, pp. 483~490.
- (7) Lee, J., Chai, J., Reitsma, P., Hodgins, J.K. and Pollard, N., 2002, "Interactive Control of Avatars Animated with Human Motion Data," *Proceedings of SIGGRAPH 2002*, Vol. 21, No. 3, pp. 491~500.
- (8) Pullen, K. and Bregler, C., 2002, "Motion Capture Assisted Animation: Texturing and Synthesis," *Proceedings of SIGGRAPH 2002*, pp. 501~508.
- (9) Kwon Dong-Soo, Yang Gi-Hun, Lee Chong-Won, Shin Jae-Cheol, et al, 2002, "KAIST Interactive Bicycle Racing Simulator: The 2nd Version with Advanced Features," *Proceedings of the IROS'02*, pp. 2961~296.
- (10) Lee Jongwhan and Han Soonhung, 2004, "Bicycle Simulator, Multiple Participants, Networked Visualization, Virtual Reality," *Transactions of the society of CAD/CAM engineers*, Vol. 9, No.3, pp. 212~219.
- (11) Yoo Byunghyun and Han Soonhung, 2005, "A Framework for VR Effects with Multi-sensory Motional Display," *Journal of the Korea Society for Simulation*, Vol. 14, No. 3, pp. 55~66.
- (12) Cha Moohyun, 2006, "A Driving Simulator Based on Motion Database and Motion Blending," KAIST Master Thesis.
- (13) SNU Movement Research Lab, <http://mrl.snu.ac.kr/>
- (14) AR-Vision, <http://www.ar-vision.com/>
- (15) Simuline, <http://www.simuline.com/>
- (16) TORCS, <http://torcs.sourceforge.net/>
- (17) KATECH, <http://www.katech.re.kr/>