

## 3D 모델링 소프트웨어 툴바 설계가 사용자 경험에 미치는 영향: Blender 4.4와 3ds Max 2025 비교 분석

최한길<sup>1</sup> · 최승관<sup>2\*</sup><sup>1</sup>서강대학교 가상융합전문대학원 석사과정<sup>2</sup>서강대학교 가상융합전문대학원 엔터테인먼트전공 주임교수 / 전임교수

### 3D Modeling Toolbar UX Comparison: Blender 4.4 and 3ds Max 2025

Hankil Choi<sup>1</sup> · Seung Kwan Choi<sup>2\*</sup><sup>1</sup>Master's Course, Graduate School of Virtual Convergence, Sogang University, Seoul 04107, Korea<sup>2</sup>Professor, Entertainment Major, Graduate School of Virtual Convergence, Sogang University, Seoul 04107, Korea

#### [요약]

본 연구는 Blender 4.4와 3ds Max 2025의 툴바 설계가 실제 작업 환경에서 사용자 경험에 미치는 영향을 체계적으로 분석하였다. 2024년 4월부터 2025년 4월까지 1년간 Blender Artists, Autodesk Forum, ScriptSpot 등 각종 주요 온라인 커뮤니티에서 수집한 사용자 피드백 총 80건과 연구자의 직접 사용 경험을 함께 검토하고 비교 분석하였다. 정량 분석 결과, 3ds Max는 복잡한 리본 메뉴 구조로 인해 전체 피드백의 77.5%가 부정적으로 평가되었으며, Blender의 부정 평가 비율은 65.0%로 상대적으로 낮게 나타났다. 3ds Max 사용자들은 핫키 시스템 충돌(32.3%)과 UI 성능 저하(25.8%)를 주요 문제로 지적하였고, Blender 사용자들은 추가 클릭 증가(26.9%)와 툴 위치 분산(23.1%)을 주된 불만 사항으로 제기하였다. Unity/Unreal 파이프라인에서 Blender의 FBX 내보내기는 안정적이었으나, 비표준 단축키 체계는 학습과 전환 과정의 부담으로 나타났다. 본 연구는 인터페이스와 학습곡선을 함께 고려해야 함을 실증한다.

#### [Abstract]

This study analyzes the influence of toolbar design of Blender 4.4 and 3ds Max 2025 on user experience in real-production environments. We collected 80 user feedback statements from online communities, including Blender Artists, Autodesk Forum, and ScriptSpot, between April 2024 and April 2025, and cross-checked them with the researcher's hands-on experience. Quantitative analysis showed that 3ds Max received 77.5% negative evaluations due to the complexity of its ribbon menu structure, while Blender's negative rate was lower at 65.0%. 3ds Max users most often reported hotkey system conflicts (32.3%) and User Interface (UI) performance degradation (25.8%) as critical issues. Blender users mainly complained about the increased extra clicks (26.9%) and dispersed tool locations (23.1%). Unity/Unreal pipeline tests showed that Blender's FBX export was relatively stable, but its non-standard shortcut system imposed a burden during learning and tool transition. This study demonstrates the importance of considering both the interface design and learning curves.

**색인어** : 사용자 경험, 3D 모델링 인터페이스, 툴바 설계, 인지부하, 정보구조**Keyword** : User Experience, 3D Modeling Interface, Toolbar Design, Cognitive Load, Information Architecture<http://dx.doi.org/10.9728/dcs.2026.27.1.29>

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 17 November 2025; Revised 09 December 2025

Accepted 02 January 2026

\*Corresponding Author; Seung Kwan Choi

Tel: 

E-mail: csk0123@sogang.ac.kr

## I. 서론

3D 콘텐츠 제작 기술은 게임, 영화, 건축, 제품 디자인 등 거의 모든 디지털 시각 분야에서 핵심 도구로 활용된다. 과거에는 모델링, 애니메이션, 렌더링을 각기 다른 소프트웨어로 수행했지만, 2000년대 중반 이후 통합 솔루션들이 등장하면서 하나의 프로그램으로 전체 파이프라인을 처리할 수 있게 되었다. 기능이 통합될수록 작업 효율은 높아졌지만, 동시에 인터페이스의 복잡도도 크게 증가했다. 사용자 관점에서 도구의 복잡성은 작업 수행을 어렵게 만드는 “실행의 격차”를 넓힌다고 Norman은 지적했다[1]. 또한 Sweller 등의 인지부하 이론에 따르면 불필요하게 복잡한 인터페이스는 사용자의 인지 자원을 소모하여 실제 과제 수행 능력을 저하시킬 수 있다[2].

기존 연구들은 주로 렌더링 속도나 폴리곤 처리량 같은 기술적 성능에 초점을 맞추었고, 정작 사용자가 매일 대면하는 툴바나 메뉴 인터페이스에 대한 연구는 상대적으로 적었다. Bhavnani와 John(1996)은 CAD 소프트웨어 사용자들이 기본적인 명령어 지식만 습득하고 효율적인 전략적 사용법으로 발전하지 못하는 현상을 보고하면서, 인터페이스 설계가 사용자의 학습과 숙련도 향상에 중요한 영향을 미친다고 지적했다[3]. Max와 Blender의 작업 경험은 사뭇 다르다. 3ds Max에서는 Q, W, E 키로 객체 선택/이동/회전을 조작하는데 이 단축키 체계가 Unity나 Unreal Engine에서도 동일하게 적용되어 SW간 전환이 수월했다. Blender는 기본 단축키가 달라 처음에는 적응에 혼란을 겪었지만 분명한 장점도 많았다. Blender는 모디파이어 패널을 중심으로 대부분 작업이 한 곳에서 이루어져 작업 동선이 짧았고, 별도 프로그램을 사용하지 않고도 내장된 Sculpt 모드로 조각 작업을 바로 수행할 수 있었다. 무엇보다 Blender에서 만든 FBX 파일을 Unity나 Unreal로 옮길 때 3ds Max에 비해 호환성이 비교적 좋았다. 이런 경험을 통해 두 프로그램의 툴바 구조 차이가 사용자 경험에 어떤 영향을 주는지 궁금해졌고, 본격적으로 비교 분석을 수행하게 되었다.

현재 툴바나 메뉴 구조처럼 사용자가 매일 접하는 인터페이스 측면을 실제 현업 사용자 관점에서 분석한 연구는 확실히 부족한 상황이다.

본 연구는 Blender와 3ds Max가 게임/건축/제품 모델링 등 동일한 실무 영역에서 서로를 대체하는 대표적 범용 3D 툴이라는 점에 주목하여 두 소프트웨어를 비교 대상으로 선정하였다. Maya나 Cinema4D는 특정 도메인(VFX·모션그래픽)에 보다 특화되어 있어, 본 연구가 다루는 범용 모델링 및 엔진 연동 중심의 비교 틀에는 적합성이 상대적으로 낮다. 이에 비해 Blender와 3ds Max는 동일한 작업을 수행하면서도 UI/툴바 구조에서 상이한 설계 철학을 갖고 있어 사용자 경험 차이를 분석하기에 타당한 조합으로 판단하였다.

이 연구에서는 Blender 4.4와 3ds Max 2025 사용자들이

온라인 커뮤니티에 남긴 실제 사용 경험을 수집하여, 두 소프트웨어의 인터페이스 설계 차이가 사용자 경험에 미치는 영향을 살펴보고자 한다.

연구 질문은 다음과 같다.

- (1) 두 소프트웨어의 툴바 구조는 어떻게 다른가?
- (2) 사용자들은 각 툴바에서 어떤 어려움이나 장점을 경험하는가?
- (3) 해당 차이점이 실제 업무환경에서 도구를 선택하는 데 어떤 시사점을 주는가?

이 연구는 3D 제작 도구 선택 시 성능뿐 아니라 인터페이스 구조와 학습 곡선을 함께 고려해야 함을 강조하고자 한다. 이를 통해 학생이나 초보자가 단순히 “업계 표준”이라는 이유만이 아니라 실제 사용 경험을 바탕으로 자신에게 맞는 툴을 합리적으로 선택하는 데 도움을 주는 것을 목표로 한다.

## II. 관련 연구

### 2-1 3D 소프트웨어의 인터페이스와 사용자 경험

3D 소프트웨어의 사용성에 대한 선행 연구는 주로 작업 효율이나 학습 난이도 같은 정량 지표에 집중되어 왔다. Bhavnani와 John은 CAD 소프트웨어의 비효율적 사용이 경험이나 훈련만으로는 해결되지 않으며, 사용자들이 명령어 지식을 넘어 전략적 지식을 습득해야 효율적인 활용이 가능하다고 밝혔다[3]. 이는 인터페이스 구조 파악이 작업 습득에서 차지하는 비중이 크다는 점을 시사한다. 한편 Norman은 도구 디자인에서 사용자의 인지적 부담을 최소화하는 것이 중요함을 강조하며[1], 복잡한 소프트웨어일수록 인터페이스 편의성이 UX에 큰 영향을 준다고 보았다.

### 2-2 정보 구조와 인지부하 이론

Sweller 등의 인지부하 이론에 따르면 인간이 한 번에 처리할 수 있는 정보량에는 한계가 있으며, 메뉴 구조가 방대하고 복잡할수록 필요한 기능을 찾는 데 불필요한 인지 자원이 소모된다[2]. Rosenfeld와 Morville은 웹사이트 정보구조 연구를 통해 사용자가 “다음 클릭에 무엇이 나올지” 예측할 수 있을 때 경험 만족도가 높아진다고 밝혔다[4]. 이러한 예측 가능성의 원리는 3D 소프트웨어 툴바 설계에도 적용될 수 있다. 인터페이스가 일관되고 예측 가능할수록 사용자는 학습과 탐색에 들이는 노력을 줄이고 본연의 창작 작업에 집중할 수 있다.

### 2-3 기존 연구의 한계 및 차별점

3D 그래픽 제작 소프트웨어 비교 연구는 많이 이루어졌지

만, 대부분 렌더링 품질이나 모델링 성능처럼 기술적 측면에 치중되어 있었다. 이중 소프트웨어 간 데이터 교환 및 상호운용성(interoperability)을 다룬 연구는 존재하나[5], 사용자가 실제 작업 과정에서 반복적으로 접하는 툴바 구조나 UI 경험을 중심으로 분석한 연구는 상대적으로 부족하다. 결과적으로 “Blender와 3ds Max의 툴바를 실제 사용자가 어떻게 느끼는가”를 체계적으로 분석한 학술 연구는 찾기 어려운 실정이다. 본 연구는 온라인에 자발적으로 게시된 사용자 피드백을 활용하여 이러한 격차를 메우고자 하며, 온라인 리뷰로부터 사용성과 UX 정보를 추출하는 방법론(예: Hedegaard & Simonsen, 2013)을 참고하였다[6].

### III. 연구 방법

#### 3-1 데이터 수집

이 연구는 실제 사용자들이 온라인 커뮤니티에 올린 생생한 의견을 수집하여 정성적/정량적으로 분석하는 방법을 채택했다. 2024년 4월부터 2025년 4월까지 1년간 Blender와 3ds Max 관련 주요 커뮤니티에서 툴바, 리본(ribbon), UI 등의 키워드로 게시글을 검색하여 데이터를 모았다. 데이터를 수집한 플랫폼은 다음과 같다:

- Blender Artists(blenderartists.org): Blender 사용자 공식 포럼
- Autodesk Forum(forums.autodesk.com): 3ds Max 공식 지원 포럼
- ScriptSpot(scriptspot.com): 3ds Max 스크립트 개발자 커뮤니티
- Chaos Forum(forums.chaos.com): V-Ray 등 Chaos 그룹 사용자 포럼(3ds Max 관련)
- Blender Developer Forum(devtalk.blender.org): Blender 개발자 커뮤니티
- Reddit(r/blender, r/3dsmax): 기타 전반적인 사용자 커뮤니티

초기에는 총 527개의 관련 게시글을 확인하였고, 이 가운데 툴바나 메뉴 구조와 직접 관련된 내용만을 선별했다. 단순 버그 보고나 하드웨어 문제에 대한 논의는 제외하였다. 최종적으로 Blender 관련 40건, 3ds Max 관련 40건, 총 80건의 사용자 피드백을 분석 대상으로 확정하였다. 또한 온라인 커뮤니티는 불완전 사례가 상대적으로 과대표집되는 경향이 있어, 본 연구는 절대적 만족도 측정이 아니라 반복적으로 나타나는 문제 유형의 ‘패턴’에 초점을 두었다.

#### 3-2 분석 방법

정량적 분류와 정성적 해석을 병행하여 의견들을 분석했

다. 우선 각 피드백이 긍정적인지 부정적인지를 판단하고, 부정적 피드백의 경우 불만 유형을 분류하여 빈도를 집계했다. 본 연구의 분석 틀은 HCI 분야의 핵심 이론들에 기반하여 세 가지 관점으로 구성하였다. (1) 기능 탐색성: Norman의 행동 7단계 모델에서 제시한 ‘실행의 걸프(Gulf of Execution)’ 개념에 근거한다[1]. Norman은 사용자가 목표를 달성하기 위해 시스템 내에서 적절한 행동을 찾아 실행하는 과정의 어려움을 실행의 걸프라 정의했다. 기능 탐색성은 이 걸프의 크기를 측정하는 지표로, 사용자가 원하는 기능을 얼마나 쉽게 찾을 수 있는지를 평가한다. (2) 일관성: Nielsen의 사용성 휴리스틱 중 ‘일관성과 표준(Consistency and Standard)’ 원칙에 근거한다[13]. Nielsen은 사용자가 동일한 맥락에서 동일한 결과를 기대하며, 이러한 예측 가능성이 무너지면 인지적 혼란이 발생한다고 설명했다. 본 연구에서는 버전 간, 작업 맥락 간 인터페이스의 일관성 유지 여부를 이 기준으로 분석했다. (3) 학습 난이도: Sweller의 인지부하 이론에 근거한다[2]. 특히 외재적 인지부하(Extraneous cognitive load) 개념을 적용하였는데, 이는 학습 내용 자체가 아닌 정보 제시 방식으로 인해 발생하는 불필요한 인지적 부담을 의미한다. 인터페이스가 복잡할수록 외재적 인지부하가 증가하여 초보자의 학습 효율이 저하된다. 이러한 이론 기반 분석 틀과 함께, Hedegaard와 Simonsen이 제안한 온라인 사용자 리뷰에서 UX 정보를 추출하는 방법론을 적용하여 분석의 체계성을 높였다[6].

한편, 연구자 본인의 두 소프트웨어 사용 경험을 함께 정리하여 사용자 피드백과 교차 검증을 시도했다. 특히 Blender와 3ds Max에서 Unity와 Unreal Engine으로 작업물을 내보내는 과정에서 겪은 차이점들을 구체적으로 기록함으로써, 툴바 및 UI 구조의 차이가 실제 게임 개발 파이프라인에 미치는 영향을 살펴보고자 했다. 이러한 혼합적 접근을 통해 정량 데이터의 객관성과 정성 데이터의 맥락 정보를 모두 확보하였다.

#### 3-3 데이터 코딩 및 측정 방법

수집된 80건의 사용자 피드백은 다음과 같은 절차를 통해 정량화하였다.

##### 1) 감정 극성(Sentiment Polarity) 분류

각 피드백은 툴바/UI에 대한 감정 극성에 따라 ‘긍정’ 또는 ‘부정’으로 분류하였다. 이러한 온라인 사용자 텍스트에서 UX 관련 정보를 정량화하는 절차는 Hedegaard & Simonsen이 제시한 방법론[6]을 참고하였다.

##### 2) 불만 유형(Complaint Category) 코딩

부정적 피드백 57건에 대해 개방 코딩(open coding) 방식으로 초기 코드를 도출한 뒤, 유사 코드를 상위 범주로 통합하는 축 코딩(axial coding)을 수행하였다. 이는 Saldaña의 질적 코딩 체계에서 제시된 절차와 동일한 방식이다[14].

3) 신뢰도 확보

단일 연구자 코딩의 한계를 보완하기 위해, 전체 데이터의 20%(16건)를 무작위로 추출하여 2주 간격으로 재코딩하는 검사-재검사(test-retest) 방식의 신뢰도 검증을 수행하였다. 이러한 절차는 질적 코딩에서 재현 가능성을 확보하기 위한 표준적 접근으로 McAlister et al.의 연구[15]를 참고하였다. 또한 본 연구에서 사용한 일치율(percent agreement)은 Halpin이 논의한 신뢰도 지표의 적용 기준에 부합한다[16]. 본 연구의 일치율은 87.5%로 수용 가능한 수준이었다.

IV. 연구 결과

4-1 정량 분석 결과

80건의 사용자 피드백을 분류한 결과, 두 소프트웨어 모두 부정적 의견이 긍정적 의견을 압도했다. Blender 관련 40건 중 긍정 14건(35.0%), 부정 26건(65.0%)이었고, 3ds Max는 40건 중 긍정 9건(22.5%), 부정 31건(77.5%)으로 집계되었다. 3ds Max가 Blender보다 12.5%포인트 높은 불만족도를 기록했다는 점에서, 툴 및 UI 설계 측면의 개선 필요성이 상대적으로 더 크다고 볼 수 있다.

표 1. 전체 피드백 분포 비교

Table 1. Comparison of overall feedback distribution

software	Total Feedback	Positive (Count)	Positive (%)	Negative (Count)	Negative (%)
Blender 4.4	40	14	35	26	65
3ds Max 2025	40	9	22.5	31	77.5

부정 피드백 57건(Blender 26, Max 31)은 불만 유형별로 재분류했다. 분류 기준은 사용자가 명시한 문제 특성, 근본 원인(구조적 문제 vs. 기술 버그), 작업 흐름 영향도를 종합 고려했다.

표 2. Blender 4.4 부정적 피드백 분류

Table 2. Blender 4.4 negative feedback categories

Issue Type	Count	Percentage (%)
Increased Additional Clicks	7	26.9
Scattered Tool Locations	6	23.1
Sidebar Overcrowding	6	23.1
Increased UI Complexity	5	19.2
Performance Degradation	2	7.7

Blender의 경우 (표 2) 사용자들이 가장 많이 언급한 문제는 “(불필요한) 추가 클릭 증가”로, 총 7건(26.9%)이었다. 다

음으로 툴/기능 위치의 분산으로 인한 혼란이 6건(23.1%), 사이드바(UI 패널)의 과밀화로 인터페이스가 비좁아졌다는 지적이 6건(23.1%) 있었다. UI 복잡성 증가에 대한 언급도 5건(19.2%)으로 나타났다. 마지막으로 성능 저하(응답 지연이나 버벅임 등)에 관한 불만이 2건(7.7%) 있었다.

표 3. 3ds Max 2025 부정적 피드백 분류(상위 불만 유형)

Table 3. 3ds Max 2025 negative feedback categories (top complaint types)

Issue Type	Count	Percentage (%)
Hotkey System Crashes	10	32.3
UI Performance Degradation	8	25.8
Ribbon Display Errors	6	19.4
UI Inconsistency	5	16.1
Scaling Issues	2	6.5

한편 3ds Max의 부정 피드백은 (표 3) “핫키 시스템 불안정(충돌)” 보고가 10건(32.3%)으로 가장 많았고, UI 자체의 성능 저하(느리거나 응답 없음 등) 보고가 8건(25.8%)으로 뒤를 이었다. 리본 메뉴 표시 오류 또는 아이콘 깨짐 현상이 6건(19.4%), UI 구성의 비일관성 지적이 5건(16.1%) 있었고, UI 스케일링 문제(고해상도 디스플레이에서 인터페이스 크기 이상 등)가 2건(6.5%)으로 파악되었다.

4-2 Blender 4.4 사용자 의견

1) Blender 4.4 유저들의 긍정적 의견

Blender에서 가장 호평받은 기능은 F3 검색 메뉴였다. “기능 이름만 알아도 바로 찾을 수 있어서 메뉴를 뒤질 필요가 없다. ‘bevel’만 쳐도 관련 옵션이 다 나온다”(Blender Artists, 2024년 5월)처럼, 정확한 메뉴 위치를 몰라도 자연어 검색으로 기능에 접근할 수 있다는 점이 강조되었다. F3 검색 메뉴는 Norman이 강조한 “지식의 내재화(knowledge in the world)” 원칙을 구현한 사례로, 사용자가 메뉴 구조를 암기하지 않아도 되도록 설계된 점이 돋보인다[1].

모디파이어(Modifier) 중심의 통합 패널 구조도 긍정 평가를 받았다. “모든 작업이 모디파이어 패널에 모여 있어서 어디서 뭘 해야 할지 헤맬 일이 없다. Array, Mirror, Bevel 같은 핵심 기능들이 한 곳에 있으니 작업 흐름이 끊기지 않는다”(Blender Artists, 2024년 8월)는 의견처럼, 오브젝트 선택 후 화면 우측 Properties 패널 내 모디파이어 탭에서 필요한 변형 작업을 순차적으로 적용할 수 있는 흐름이 효율적으로 평가되었다. 특히 반복 패턴이 많은 작업에서 Array → Mirror → Bevel을 한 패널에서 연속 적용할 수 있어 작업 동선과 인지부하를 동시에 줄여 준다는 평가가 있었다.

Blender 내장 Sculpt 모드의 통합성도 편의성 측면에서 주목받았다. “ZBrush나 Mudbox 없이 Tab 키만 눌러 Sculpt 모드로 전환하면 바로 디테일 작업을 이어갈 수 있

다. 파일 내보내기/불러오기 과정이 없으니 시간이 절약된다”(Blender Developer Forum, 2024년 10월)는 평가가 대표적이다.

## 2) Blender 4.4 유저들의 부정적 의견

### • 추가 클릭 증가 문제(26.9%, 7건)

Blender 4.4 부정 피드백 중 가장 높은 비율을 차지한 문제다. “이전 버전에서는 단축키 하나로 끝났던 기능이 4.x부터는 메뉴를 열고 옵션을 선택하는 2~3단계를 거쳐야 한다”(Blender Artists, 2024년 9월)는 지적이 대표적이다.

UV 편집 모드에서 Snap 옵션 활성화가 대표 사례다. “이제 상단 바 Snapping 드롭다운을 클릭하고 ‘Snap to Vertex/Edge/Face’ 중 하나를 선택해야 한다. 예전엔 Shift+Tab으로 토글했는데, 지금은 매번 마우스로 메뉴를 찾아야 한다. 하루 작업 중 수십 번 반복하다 보면 시간 낭비가 체감된다”(Blender Developer Forum, 2024년 11월)는 구체적 불편 사례가 보고되었다. 단축키 하나로 즉각 실행되던 기능을 메뉴 계층 안으로 숨기면 물리적 이동거리와 인지적 탐색 시간이 동시에 증가한다. 3D 모델링처럼 반복 조작이 많은 작업에서는 누적 시간 손실이 상당하다.

Geometry Nodes에서도 비슷한 문제가 제기되었다. “Add Node 단축키가 Shift+A에서 변경되었는데, Add 메뉴가 화면 중앙이 아니라 마우스 커서 위치에 나타난다. 커서가 화면 구석에 있으면 메뉴가 잘려서 스크롤해야 한다”(Blender Artists, 2024년 12월)는 불만이 있었다. UI 일관성 부족과 컨텍스트 메뉴의 잘못된 위치 지정이 복합적으로 작용한 문제로, 원하는 노드를 찾기 위해 추가 시각 탐색과 스크롤 동작을 수행해야 한다. 캐릭터 리깅처럼 수백 개의 본과 컨스트리인트를 조작하는 작업에서 매번 메뉴를 열어야 하는 설계는 작업자의 몰입을 방해하며, 추가 클릭 증가는 단순 시간 손실을 넘어 창작 경험의 질적 저하로 이어질 수 있다.

### • 툴/기능 위치 분산 문제(23.1%, 6건)

Blender Animation Studio 개발자들조차 내부 토론에서 “툴 설정이 Dope Sheet, Topbar, Sidebar, Properties Editor 등 화면 곳곳에 흩어져 있어 일관성이 떨어진다”며 “툴이 all over the place”라는 표현도 사용했다(Blender Developer Forum, 2024년 6월).

‘Smooth’ 기능이 대표적 혼란 사례다. “똑같은 Smooth 기능이 세 곳에 중복된다. Modifier 패널의 Smooth Modifier, Mesh 메뉴의 Shade Smooth, 우클릭 컨텍스트 메뉴의 Smooth. 각각 어떤 차이가 있는지 초보자는 전혀 알 수 없고, 숙련자도 헷갈린다”(Blender Artists, 2025년 1월)는 지적이 있었다. Shade Smooth는 노멀 벡터 계산 방식을 변경하는 비파괴적 명령인 반면, Smooth Modifier는 지오메트리를 실제로 변형하는 모디파이어라 완전히 다른 기능인데, 동일한 이름과 분산된 위치로 인해 시행착오를 통해 차이를 학습해

야 한다.

Animation 워크스페이스의 키프레임 보간 설정도 문제가 제기되었다. “보간 설정이 Timeline 하단 아이콘, Graph Editor의 Key 메뉴, Properties 패널의 Active Keyframe 섹션 세 곳에 분산되어 있다. 어느 곳에서 설정하든 같은 결과를 내지만, 왜 세 곳에 나눠놓았는지 모르겠다”(Blender Developer Forum, 2024년 8월)는 의견이 있었다. 기능 배치가 예측 가능하지 않으면 사용자는 매번 정신적 지도를 재구성해야 하며, 불필요한 정보 탐색에 인지 자원이 소모되면 실제 창작 작업에 투입할 여력이 줄어든다. “이 기능이 어디 있을 것 같다”는 직관이 통하지 않아 탐색적 학습에 과도한 시간을 소비하게 된다.

### • 사이드바 과밀화 문제(23.1%, 6건)

“화면 오른쪽 Properties 패널에 Modifier, Material, Particle, Physics, Constraint 등 수십 개 탭이 세로로 늘어서 있는데, 1080p 모니터에서는 스크롤을 계속 내려야 원하는 섹션을 찾을 수 있다. 4K 듀얼 모니터 환경에서도 답답하다”(Reddit r/blender, 2024년 7월)는 불만이 제기되었다. 모디파이어가 10개 이상 쌓인 복잡한 모델에서는 시각적 탐색 시간이 급격히 증가했다. “Add-on을 여러 개 설치하면 사이드바(N-Panel)에 Add-on 전용 탭들이 추가되어 기본 Tool, Item, View 탭까지 밀려나 찾기 힘들어진다. UI 공간 관리가 전혀 안 되고 있다”(Blender Artists, 2024년 12월)는 비판도 있었다. 수십 개의 탭과 옵션이 한 화면에 밀집되면 선택지가 많아져 의사결정 시간이 증가한다. 화면 공간 분배 관점에서 Blender는 기능 추가에 따라 단순히 패널을 누적해온 것으로 보인다. 체계적인 UI 우선순위 없이 모든 기능을 동등하게 나열하면, 자주 쓰는 기능과 거의 안 쓰는 기능을 구분하기 어렵다.

### • UI 복잡성 증가 문제(19.2%, 5건)

“Blender 3.x 대비 4.4는 메뉴 깊이가 깊어지고 옵션이 세분화되면서 단순한 작업도 복잡하게 느껴진다”(Blender Artists, 2024년 10월)는 의견이 대표적이다.

Subdivision Surface 모디파이어 설정에서 변화가 두드러진다. “예전엔 ‘Levels Viewport’와 ‘Levels Render’ 두 슬라이더만 있었는데, 4.4부터는 ‘Use Limit Surface’, ‘Optimal Display’, ‘Use Creases’ 등 고급 옵션이 기본 패널에 함께 노출된다. 초보자는 이게 뭔지 모르겠고, 숙련자도 대부분 기본값으로 둔다. 굳이 왜 기본 화면에 다 보여주나?”(Reddit r/blender, 2024년 11월)는 비판이 제기되었다.

교육 관점의 문제도 있다. “초보자 튜토리얼을 따라하려 해도 UI 구조가 버전마다 달라져서 강사가 말하는 버튼을 찾지 못해 혼란스럽다. ‘Shade Smooth’ 버튼이 Blender 3.6에서는 좌측 툴바에 있었는데 4.2부터는 우클릭 메뉴로 옮겨갔다. 튜토리얼과 실제 UI가 다르니 배우기 어렵다”(Reddit r/blender, 2025년 2월)는 불만도 있었다.

메뉴 계층이 깊어지고 선택지가 많아질수록 “이 옵션을 켜면 무슨 일이 일어날까?”를 예측하기 위해 추가적인 정신적 노력을 기울여야 하며, 실제 모델링이나 창작 작업에 쓸 인지 자원을 소모한다. UI 디자인에서는 “사용자가 필요할 때만 고급 옵션을 보여주고, 기본 화면은 단순하게 유지한다”는 점진적 노출 원칙이 있는데, Blender 4.4는 초보자가 거의 사용하지 않는 고급 파라미터까지 기본 패널에 노출하여 시각적 잡음을 증가시켰다.

• 성능 저하 문제(7.7%, 2건)

상대적으로 보고 건수는 적었지만, 경험한 사용자에게는 치명적이었다. “Blender 4.1에서 Outliner 패널을 펼칠 때마다 1~2초씩 멈춘다. 씬에 오브젝트가 5,000개 이상이면 UI 반응성이 현저히 떨어진다. 3.6에서는 문제없었는데 4.x로 올리면서 뭔가 최적화가 안 된 것 같다”(Blender Developer Forum, 2024년 8월)는 보고가 있었다.

또 “Modifier 패널에서 파라미터 슬라이더를 조정할 때 끊기는 느낌이 든다. 예전엔 실시간으로 부드럽게 반영됐는데, 지금은 슬라이더를 움직이면 0.5초 지연 후 화면이 업데이트 된다”(Blender Artists, 2025년 3월)는 불만도 제기되었다.

응답 지연은 작업자의 몰입을 방해한다. 작업 중 예측 불가능한 중단은 몰입 상태를 깨뜨려 창의적 사고를 저해하며, 3D 작업은 실시간 피드백이 중요한데 파라미터 조정 후 결과를 즉시 확인하지 못하면 시행착오 학습이 비효율적이 된다.

일반적으로 시스템 응답 시간이 1초를 넘으면 사용자는 “뭔가 잘못됐다”고 인식하며 주의가 분산되는데, Blender의 1~2초 UI 지연은 이 임계값을 초과하여 부정적 사용자 경험을 유발한다.

4-3 3ds Max 2025 사용자 의견

1) 3ds Max 유저들의 긍정적 의견

3ds Max 긍정 피드백의 핵심은 Q/W/E로 대표되는 업계 표준 단축키 체계였다. “Max의 Q/W/E 단축키는 정말 편하다. Unity, Unreal, Maya 모두 같은 키를 쓰니까 프로그램을 바꿔도 헛갈리지 않는다. 아침에 Max로 모델링하고 오후에 Unreal로 넘어가도 근육 기억이 그대로 작동한다”(Autodesk Forum, 2024년 5월)는 평가처럼, 여러 툴을 병행하는 작업 환경에서 일관된 조작 체계가 주는 장점이 부각되었다.

인지심리학에서 전이 학습이란 한 맥락에서 학습한 기술이 다른 맥락에서도 적용되는 현상을 말한다[12]. 3ds Max의 Q/W/E 단축키가 Unity/Unreal과 동일하면 재학습 비용 없이 소프트웨어 간 전환이 가능하며, 여러 프로그램을 오가는 게임 개발 환경에서 전환 비용을 줄여 파이프라인 효율을 높인다.

2) 3ds Max 유저들의 부정적 의견

• 핫키 시스템 불안정 문제(32.3%, 10건)

3ds Max 2025의 부정 피드백 중 가장 많았던 문제다. “Customize → Customize User Interface → Keyboard 탭을 열면 일관되게 크래시가 발생한다. ‘문제가 발생하여 3ds Max를 종료해야 합니다’라는 오류 알림이 뜬다. 단축키를 커스터마이징할 수 없어 작업을 못 한다”(Autodesk Forum, 2024년 4월)는 치명적 버그 보고가 있었다.

커뮤니티에서는 “핫키 편집기를 열 때마다 MaxScript 에러 로그에 ‘undefined property’ 메시지가 수십 줄씩 쌓인다. Autodesk가 테스트는 했나?”(ScriptSpot, 2024년 6월)는 비판도 나왔다.

시스템 불안정은 신뢰성 측면에서 사용자 경험을 심각하게 훼손한다. 도구가 예측 가능하지 않으면 사용자는 불안을 느끼고 도구를 신뢰하지 못하는데, 단축키는 전문 사용자의 핵심 생산성 도구라는 점에서 이것이 무작위로 작동하지 않거나 프로그램을 다운시킨다면 작업 자체를 포기하게 된다.

“클라이언트 미팅 2시간 전에 Max가 다운되어 파일을 복구하느라 밤샘 작업을 했다. Autodesk에 환불 요청했다”(Reddit r/3dsmax, 2024년 7월)는 토로까지 나왔다.

• UI 성능 저하 문제(25.8%, 8건)

“리본 메뉴를 클릭하면 0.5~1초 지연 후 드롭다운이 펼쳐진다. Modeling 탭에서 Rendering 탭으로 전환할 때도 화면이 잠깐 멈춘다. 3ds Max 2024에서는 없던 현상인데 2025부터 UI가 무겁게 느껴진다”(Autodesk Forum, 2024년 9월)는 보고가 있었다.

“Modifier 패널에서 Turbosmooth 이터레이션 값을 변경하면 UI가 3~5초간 얼어붙는다. 뷰포트는 업데이트되는데 패널만 응답이 없다”(Chaos Forum, 2024년 11월)는 불만도 제기되었다.

UI 응답 속도는 사용자 만족도의 핵심 지표다. 보통 시스템 응답 시간이 0.1초 이하면 즉각적으로 느껴지고, 1초 이하면 허용 가능하며, 10초 이상이면 다른 작업으로 주의를 돌리게 된다. 3ds Max의 0.5~1초 지연은 허용 가능 범위의 상한선에 걸쳐 있어, 반복 작업 시 누적 불만이 쌓인다.

• 리본 메뉴 표시 오류 문제(19.4%, 6건)

“고해상도 모니터(4K)에서 리본 아이콘이 깨져 보인다. 특히 Modeling 탭의 ‘Edit Poly’ 섹션 아이콘들이 픽셀화되어 무엇을 의미하는지 알아보기 힘들다”(Autodesk Forum, 2024년 8월)는 지적이 있었다. “다크 모드를 적용하면 일부 리본 버튼의 텍스트가 검은 배경에 검은 글씨로 나와 읽을 수 없다. UI 테스트를 전혀 안 한 것 같다”(Reddit r/3dsmax, 2024년 10월)는 불만도 제기되었다.

시각적 일관성 부족과 고해상도 디스플레이 대응 실패를 보여주는 사례다. UI 아이콘은 시각적 단서로서 사용자가 기

능을 빠르게 식별하도록 돕는데, 아이콘이 깨지면 텍스트 라벨을 읽어야 하므로 인지 속도가 느려진다. 3D 작업자는 아이콘의 형태를 기억하여 마우스로 즉시 클릭하는데, 아이콘이 픽셀화되면 시각적 기억이 무용지물이 된다.

• UI 구성 비일관성 문제(16.1%, 5건)

“플러그인을 설치하면 메뉴 에디터에는 보이는데 리본 패널에는 안 나타난다. 메시지 로그에 ‘CreateMenuAction failed’ 오류만 뜬다. 동일한 설정의 다른 PC에서는 리본 통합이 잘 되는데 내 PC에서만 문제가 지속된다”(Autodesk Forum, 2024년 7월)는 보고도 있었다.

3ds Max 2025에서 기존 menuMan 스크립트 API가 제거되며 사용자 메뉴 커스터마이징 방식이 크게 변경된 점이 혼란을 키웠다. ScriptSpot 등 개발자 커뮤니티에서는 “2024까지는 MaxScript로 menuMan을 통해 메뉴를 수정했는데, 2025부터 이 기능이 사라져 기존 스크립트를 다 새로 배워야 한다. 새 메뉴 시스템은 이전 것보다 훨씬 복잡해 보인다”(ScriptSpot, 2024년 8월)는 불만이 나왔다.

하위 호환성 부족과 문서화 미비를 보여준다. 소프트웨어 공학에서 API 변경 시 Deprecation Warning을 통해 사용자에게 사전 통지하고 마이그레이션 가이드 제공이 표준이지만, 3ds Max 2025는 menuMan API를 갑작스럽게 제거하여 고급 사용자의 워크플로우를 파괴했다.

스튜디오 파이프라인에서는 수년간 축적된 커스텀 스크립트와 플러그인이 핵심 자산인데, 이것이 신 버전에서 작동하지 않으면 업그레이드를 포기하거나 수개월의 재개발 비용을 감수해야 한다. “3ds Max 2024로 돌아갔다. 2025는 안정화될 때까지 기다린다”(CGSociety, 2024년 9월)는 이야기까지 나왔다.

• FBX 내보내기 호환성 문제

3ds Max 사용자들이 특히 심각하게 지적한 문제는 Unity/Unreal로의 FBX 내보내기 호환성이었다. “Path Deform 모디파이어를 적용한 애니메이션을 FBX로 수출하면 키프레임이 모조리 사라져 Unity에서 사용할 수가 없다. 2024에서도 같은 문제였는데 2025에서도 안 고쳐졌다”(Polycount 포럼, 2024년 6월)는 보고가 있었다.

“Physical Camera를 FBX로 내보내면 Unreal에서 FOV와 위치가 완전히 틀어진다. Maya에서는 문제없이 되는데 왜 Max만 안 되나?”(Autodesk Forum, 2024년 10월)는 불만도 나타났다.

파이프라인 통합성 실패를 보여주는 사례다. 게임 개발 워크플로에서는 모델링 툴 → 게임 엔진으로의 원활한 데이터 전송이 필수인데, 3ds Max의 FBX 익스포터는 Autodesk 자체 포맷에 최적화되어 있고 업계 표준인 FBX에는 상대적으로 소홀한 것으로 보인다.

매번 FBX 내보내기 오류를 체크하고 수동 수정해야 한다면 인터레이션 속도가 급격히 저하되어 빠른 프로토타이핑을 저해하는 요소다.

4-4 Blender & 3ds Max 툴바 설계 비교 분석

Blender 4.4는 좌측 툴바와 우측 Properties(모디파이어 중심) 패널을 핵심 작업축으로 두고, 3ds Max 2025는 상단 Ribbon과 우측 Command Panel을 중심으로 기능을 분류·배치한다. 즉, Blender는 검색(F3)과 패널 중심 흐름이 강하고, 3ds Max는 탭 기반 계층(리본) 탐색의 비중이 크다. 이하 비교는 이러한 기본 레이아웃 차이를 전제로 기능 탐색성·일관성·학습 난이도를 해석한다.

1) 기능 탐색성 비교

두 소프트웨어의 기본 인터페이스 구조는 Blender가 ‘좌측 툴 셀프-우측 모디파이어 패널’ 중심인 반면, 3ds Max는 ‘상단 리본-우측 커맨드 패널’ 구조를 사용한다는 점에서 시각적 배치 차이가 존재한다. Blender의 F3 검색 메뉴는 자연어 기반 검색을 지원하여 기능의 정확한 이름을 몰라도 유사 키워드로 찾을 수 있다. “smooth”를 검색하면 Smooth Shading, Smooth Modifier, Smooth Vertices 등 관련 기능이 모두 나열된다. 정보 검색 관점에서 Recall(재현율)이 높은 설계다.

반면 3ds Max의 리본 메뉴는 계층적 분류에 의존하는데, “이 기능이 어느 카테고리에 속할까?”를 예측해야 한다. ‘Symmetry’ 기능이 Modeling 탭에 있을지 Modify 탭에 있을지 초보자는 알 수 없다. Rosenfeld & Morville이 지적한 “예측 가능성 부족” 문제와 직결된다[4].

Blender 사용자 중 35.0%가 긍정 평가를 했으나 3ds Max는 22.5%에 그쳤는데, 기능 탐색의 용이성 차이가 전체 만족도에 영향을 준 것으로 해석된다.

2) 단축키 표준화 비교

3ds Max의 Q/W/E 체계는 업계 표준으로 Unity, Unreal, Maya, MotionBuilder 등 도구 전환 비용을 최소화하여 여러 프로그램을 오가는 작업 환경에서 유리하다.

표 4. Blender vs 3ds Max 주요 변환 기능 단축키 및 Unity/Unreal 대비

Table 4. Key transformation shortcuts: Blender vs. 3ds Max compared with Unity/Unreal

Function	Blender Key	3ds Max Key	Unity/Unreal Key
Select	Mouse Click	Q	Q
Move	G	W	W
Rotate	R	E	E
Scale	S	R	R

Blender의 G/R/S 단축키는 직관적(Grab/Rotate/Scale)이지만 업계 표준과 다르다. Industry Compatible 키맵을 제공하나, 모든 기능이 완벽히 매핑되지 않아 부분적 호환에 그친다.

게임 개발자가 하루에 Max → Unity → Substance → Unreal을 오간다면, 3ds Max의 단축키 일관성은 근육 기억 유지에 유리하다. Blender 사용자는 매번 키맵을 의식적으로 전환해야 하므로 인지적 부담이 증가한다.

### 3) Unity/Unreal 파이프라인 호환성 비교

Blender의 FBX 내보내기는 상대적으로 안정적이다. 180도 회전 문제나 경고 메시지가 간헐적으로 발생하지만, 데이터 유실은 드물다. UV, Normal, Vertex Color 같은 메시 속성이 대부분 보존된다.

3ds Max는 Path Deform, Skin Wrap 같은 고급 모디피어의 FBX 변환이 불안정하다. 키프레임 유실, 카메라 FOV 오류 등 치명적 데이터 손실이 발생하여 사용자가 수동적으로 수정해야 한다.

Blender는 오픈소스 커뮤니티 주도로 모듈식 설계를 채택했다. 기능 추가가 자유롭지만 일관성 유지가 어렵다. FBX 등 오픈 포맷 호환성을 우선시하며 커뮤니티 피드백에 빠르게 대응한다.

3ds Max는 Autodesk의 상업 소프트웨어로 레거시 코드의 의존도가 높다. 20년 이상 누적된 코드베이스로 인해 새 UI 시스템 통합이 복잡하다. Autodesk 생태계 내 폐쇄적 통합을 선호하며 FBX는 부차적 기능으로 취급한다.

### 4) 인지부하 이론 관점의 종합 해석

본 연구에서 도출된 부정적 피드백은 Sweller 등의 인지부하 이론으로 해석할 수 있다[2]. 인지부하는 내재적 부하(과제 자체의 복잡성)와 외재적 부하(정보 제시 방식의 비효율성)로 구분되는데, 두 소프트웨어의 UI 문제는 대부분 외재적 인지부하에 해당한다.

Blender의 ‘추가 클릭 증가’(26.9%)는 Fitts의 법칙[7] 관점에서 해석된다. 타겟까지 거리가 멀고 타겟 크기가 작을수록 포인팅 시간이 길어지므로, 단축키에서 메뉴 계층으로 기능이 이동하면 물리적·인지적 비용이 동시에 증가한다. ‘툴 위치 분산’(23.1%)은 Norman이 강조한 ‘개념 모델의 일관성’ 부재에 해당한다[1]. 동일한 ‘Smooth’ 기능이 세 곳에 중복 배치되면 사용자는 ‘어느 것이 맞는가?’라는 불필요한 판단에 인지 자원을 소모한다. ‘사이드바 과밀화’(23.1%)는 Hick의 법칙[9]과 관련되며, 선택지가 많을수록 의사결정 시간이 증가한다. ‘UI 복잡성 증가’(19.2%)는 점진적 노출 원칙[10] 위반으로, 초보자에게 불필요한 고급 옵션 노출이 외재적 인지부하를 가중시킨다.

3ds Max의 ‘핫키 시스템 충돌’(32.3%)은 Norman이 강조한 시스템 신뢰성 문제다[1]. 도구가 예측 가능하지 않으면 사용자는 불안함을 느끼고 도구를 신뢰하지 못한다. ‘UI 성능 저하’(25.8%)는 Nielsen의 응답 시간 연구[11]와 관련되며, 1초 이상의 지연은 사용자 주의를 분산시킨다. ‘리본 메뉴 표시 오류’(19.4%)와 ‘UI 구성 비일관성’(16.1%)은 시각적 피드백의 신뢰성 저하로, 사용자가 인터페이스를 통해 시스템

상태를 정확히 파악하지 못하게 만든다. ‘FBX 내보내기 호환성’ 문제는 파이프라인 통합성 실패로, 작업 결과물의 예측 가능성을 저해한다.

두 소프트웨어의 ‘성능 저하’ 문제(Blender 7.7%, 3ds Max 25.8%)는 Csikszentmihalyi의 몰입 이론[8]으로 해석할 수 있다. 작업 중 예측 불가능한 중단은 몰입 상태를 깨뜨려 창의적 사고를 저해하며, 3D 작업은 실시간 피드백이 중요하므로 UI 지연은 시행착오 학습을 비효율적으로 만든다.

한편, 3ds Max의 긍정적 평가 요인인 Q/W/E 단축키 체계는 Perkins와 Salomon이 제시한 학습 전이(transfer of learning) 개념[12]으로 설명된다. 한 맥락에서 습득한 기술이 유사한 환경에서 재학습 없이 적용되는 ‘근접 전이(near transfer)’가 발생하려면 두 환경 간 유사성이 높아야 하는데, 3ds Max의 단축키 체계는 Unity, Unreal, Maya와의 높은 유사성을 통해 전이 비용을 최소화한다.

결론적으로, 3ds Max의 77.5% 부정 평가율은 기술적 불안정성이, Blender의 65.0% 부정 평가율은 구조적 분산 문제가 외재적 인지부하를 높인 결과로 해석된다. 두 소프트웨어 모두 ‘외재적 부하 최소화’ 원칙을 충분히 구현하지 못하고 있으며, 이는 향후 UI 개선의 핵심 방향이 되어야 할 것이다.

## V. 논 의

분석 결과를 종합하면, 두 소프트웨어 툴바 설계의 가장 큰 차이는 “예측 가능성”과 “일관성”에서 드러났다. Blender는 F3 검색 기능 덕분에 처음 사용하는 사람도 비교적 쉽게 필요한 기능을 찾을 수 있었지만, Blender Animation Studio 개발팀이 지적한 대로 툴 설정이 Dope Sheet, Topbar, Sidebar, Properties Editor 등 여러 영역에 분산되어 UI 일관성 측면에서는 아쉬움이 있었다. 반면 3ds Max는 리본 기반 인터페이스가 길보기에는 체계적으로 정돈된 느낌을 주지만, 실제로는 중첩된 메뉴 구조와 탭 간 분산으로 인해 사용자가 기능 위치를 예측하기 어렵고 학습에 상당한 시간이 소요되었다.

이러한 결과를 Sweller 등의 인지부하 이론으로 해석하면, 두 소프트웨어의 부정적 경험은 대부분 외재적 인지부하에서 기인한다[2]. 외재적 인지부하란 과제 자체가 아닌 정보 제시 방식의 비효율성에서 발생하는 불필요한 인지 부담을 말한다. 3ds Max의 경우 사용자가 ‘Symmetry’ 기능이 어느 탭에 있는지 예측할 수 없어 메뉴 탐색 자체에 인지 자원을 소모하게 되며, 이것이 77.5%라는 높은 부정 평가율의 원인으로 보인다. 반면 Blender의 F3 검색기능은 메뉴 구조 암기 없이 기능에 접근하게 하여 외재적 부하를 줄이는 설계다. 다만 Blender의 ‘툴 위치 분산’ 문제(23.1%)는 동일 기능이 여러 곳에 중복 배치되어 ‘어느 것이 맞는가?’라는 불필요한 판단을 요구한다는 점에서 외재적 인지부하에 해당한다.

단축키 표준화 문제는 현업에서 예상보다 훨씬 중요하게 작용했다. 3ds Max의 Q/W/E 단축키 체계는 Unity, Unreal Engine, Maya와 완전히 일치하여 여러 프로그램을 오갈 때 자연스러운 전환 경험을 제공했다. 이는 Perkins와 Salomon이 제시한 학습 전이 개념으로 설명되는데, 유사한 환경 간 이동 시 재학습 비용이 최소화된다[12]. 게임 개발자가 하루에도 3ds Max → Substance Painter → Unity 등 여러 툴을 오가는 환경에서 동일한 단축키 체계는 근육 기억을 유지시켜 인지적 전환 비용을 최소화한다. Blender도 Industry Compatible 키맵을 제공하나 완벽한 호환은 아니어서, 팀 작업이나 교육 환경에서는 여전히 적응 시간이 필요하다.

Unity/Unreal Engine과의 파이프라인 호환성 측면에서는 Blender의 FBX 내보내기가 전반적으로 더 안정적인 것으로 나타났다. Blender에서도 180도 회전이나 축 뒤집힘 같은 간헐적 문제가 보고되었지만, 3ds Max에서는 Path Deform 모디파이어 사용 시 키프레임 전체가 유실되거나 Physical Camera의 FOV가 왜곡되는 등 치명적 데이터 손실이 빈번했다. 때문에 3ds Max에서 작업한 파일을 Blender로 불러와 재수출하는 우회 경로를 구축해야 했는데, 이는 명백히 비효율적이다.

3ds Max 2025 버전에서 menuMan 스크립트 API가 예고 없이 제거되며 기존 스크립트 호환성 문제가 발생한 점도 주목할 만하다. 소프트웨어 공학에서 API 변경 시 Deprecation Warning을 통해 사용자에게 사전 통지하고 마이그레이션 가이드를 제공하는 것이 표준임에도 불구하고, Autodesk는 이를 이행하지 않아 고급 사용자의 워크플로를 파괴했다. Blender 역시 버전 업데이트마다 UI 변경으로 인한 혼란이 발생했다. Shade Smooth 버튼이 Blender 3.6에서는 좌측 툴바에 있었는데 4.2부터는 우클릭 메뉴로 이동하여, 기존 튜토리얼을 따라하는 초보자들이 30분 이상 헤매는 사례가 빈번했다. 두 소프트웨어 모두 Nielsen이 강조한 ‘일관성과 표준’ 원칙 유지에 어려움을 겪고 있으며[13], 이는 장기 사용자에게 큰 시행착오 비용을 요구한다.

Blender와 3ds Max의 설계 철학 차이가 이러한 문제의 근본 원인으로 작용한다. Blender는 오픈소스 커뮤니티 주도로 기능 추가가 자유롭지만 일관성 유지가 어렵고, FBX 등 오픈 포맷 호환성을 우선시한다. 반면 3ds Max는 누적된 레거시 코드로 인해 새 UI 시스템 통합이 복잡하며, Autodesk 생태계 내 폐쇄적 통합을 선호하여 FBX는 부차적 기능으로 취급된다. 이처럼 툴바 UI의 사용성뿐만 아니라 작업 파이프라인 전체의 설계와 안정성이 사용자 경험에 큰 영향을 미친다. 특히 게임 개발처럼 빠른 이터레이션이 핵심인 분야에서는 모델링 툴과 게임 엔진 간 원활한 데이터 전송이 원활해야 하며, 호환성 문제는 프로젝트 전체 생산성을 저해한다. 그래픽 제작 툴에서는 인터페이스가 작업자의 사고 흐름을 방해하지 않아 사용자가 도구를 의식하지 않고 작업물에 집중할 수 있어야 한다. 본 연구에서 확인된 주요 불편 요소들은 이러한 몰입을 무너뜨리는 외재적 인지부하로 해석된다.

## VI. 결 론

본 연구는 Blender 4.4와 3ds Max 2025의 툴바 인터페이스가 사용자 경험에 미치는 영향을 80건의 실제 사용자 피드백을 통해 비교 분석했다. 3ds Max는 부정적 평가 77.5%로 Blender(65.0%)보다 사용자 불만이 높았고, 핫키 시스템 충돌(32.3%)과 UI 성능 저하(25.8%) 때문이었다. Blender는 긍정 평가 비율이 더 높았지만(35.0%), 추가 클릭 증가(26.9%)와 툴 위치 분산(23.1%)이 한계로 지적되었다.

초보자 및 교육 환경에서는 Blender의 완만한 학습곡선이 유리하다. F3 검색 기능과 통합된 모디파이어 패널 덕분에 입문자가 빠르게 생산성에 도달할 수 있다.

게임 개발 파이프라인에서는 Blender의 비교적 안정적인 FBX 호환성이 이터레이션 속도 향상에 유리하다.

전문 스튜디오나 기존 파이프라인이 확립된 환경에서는 3ds Max의 업계 표준 호환성이 여전히 중요하다. Q/W/E 단축키 체계가 Unity, Unreal, Maya와 일치하여 여러 툴을 오가는 작업자의 인지적 전환 비용을 최소화한다. 대규모 프로젝트에서 수십 명의 아티스트가 협업할 때, 동일한 단축키 체계는 커뮤니케이션 비용을 줄이고 신규 인력 온보딩을 수월하게 만든다. 다만 초기 학습에 더 많은 시간과 비용을 투자해야 한다는 점을 감수해야 한다.

프리랜서나 개인 창작자에게는 Blender의 올인원 통합 환경이 효율적이다. 무료이면서도 모델링부터 렌더링, 조각, 애니메이션까지 한 프로그램 내에서 모두 처리할 수 있어 작업 전환 비용이 낮다. ZBrush나 Substance Painter 같은 유료 보조 프로그램 없이도 대부분의 작업을 완결할 수 있어, 예산이 제한된 창작자에게 현실적인 선택지가 된다.

이 연구는 몇 가지 한계를 갖는다. 우선, 데이터 소스가 주로 영어권 온라인 커뮤니티에 집중되어 다른 지역 사용자의 의견을 충분히 반영하지 못했으며, 인터넷 게시글의 특성상 부정적 경험이 더 활발히 공유되어 피드백이 부정 편향되었을 가능성이 있다. 또, 연구자 개인의 사용 경험이 게임 개발 파이프라인 중심이어서 건축 비주얼라이제이션이나 영화 VFX 같은 다른 분야에서는 평가가 달라질 수 있다. 마지막으로 Blender와 3ds Max는 버전 업데이트에 따라 UI/툴바 배치가 변동될 수 있으므로, 본 결과는 Blender 4.4와 3ds Max 2025 시점의 UX를 반영한 분석으로 해석되어야 한다.

향후 연구에서는 동일 과제 기반의 통제 실험을 통해 작업 완료 시간, 클릭 횟수, 오류 빈도 등 객관적 사용성 지표를 측정하여, 본 연구의 커뮤니티 기반 결과(문제 유형 분포)를 교차 검증할 필요가 있다.

“최고의 3D 프로그램”은 정해진 것이 아니라 사용자 개인의 목적, 작업 환경, 협업 구조, 예산 등에 따라 달라진다. 중요한 것은 각 도구의 장단점을 이해하고 자신의 맥락에 맞는 선택을 하는 일이다.

본 연구는 기술적 성능 중심의 기존 연구와 달리, 툴바와

메뉴 구조가 실제 작업 경험에 미치는 영향을 실증 분석했다는 점에서 의의가 있다. 향후 3D 소프트웨어 개발자들이 사용자 중심 설계에 관심을 기울이고, 학생이나 초보자들이 실제 사용 경험을 바탕으로 합리적인 도구 선택을 할 수 있기를 기대한다.

### 감사의 글

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 ‘메타버스 융합대학원’ 과제(RS-2022-00156318)와, 문화체육관광부 및 한국콘텐츠진흥원의 ‘메타버스에서 저작권 보호 및 이용 활성화 기술 개발’ 과제(RS-2023-00219237)의 지원을 받아 수행되었다.

### 참고문헌

[1] D. A. Norman, *The Design of Everyday Things: Revised and Expanded Edition*, New York, NY: Basic Books, 2013.

[2] J. Sweller, P. Ayres, and S. Kalyuga, *Cognitive Load Theory*, New York, NY: Springer, 2011. <https://doi.org/10.1007/978-1-4419-8126-4>

[3] S. K. Bhavnani and B. E. John, “Exploring the Unrealized Potential of Computer-Aided Drafting,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '96)*, Vancouver: Canada, pp. 332-339, April 1996. <https://doi.org/10.1145/238386.238538>

[4] L. Rosenfeld, P. Morville, and J. Arango, *Information Architecture: For the Web and Beyond*, 4th ed. Sebastopol, CA: O’Reilly Media, 2015.

[5] W. Gielingh, “An Assessment of the Current State of Product Data Technologies,” *Computer-Aided Design*, Vol. 40, No. 7, pp. 750-759, 2008.

[6] S. Hedegaard and J. G. Simonsen, “Extracting Usability and User Experience Information from Online User Reviews,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*, Paris: France, pp. 2089-2098, April 2013. <https://doi.org/10.1145/2470654.2481286>

[7] P. M. Fitts, “The Information Capacity of the Human Motor System in Controlling the Amplitude of Movement,” *Journal of Experimental Psychology*, Vol. 47, No. 6, pp. 381-391, 1954. <https://doi.org/10.1037/h0055392>

[8] M. Csikszentmihalyi, *Flow: The Psychology of Optimal Experience*, New York, NY: Harper & Row, 1990.

[9] W. E. Hick, “On the Rate of Gain of Information,” *Quarterly Journal of Experimental Psychology*, Vol. 4, No. 1, pp.

11-26, March 1952. <https://doi.org/10.1080/17470215208416600>

[10] W. Lidwell, K. Holden, and J. Butler, *Universal Principles of Design*, Beverly, MA: Rockport Publishers, 2010.

[11] J. Nielsen, *Usability Engineering*, San Francisco, CA: Morgan Kaufmann, 1993.

[12] D. N. Perkins and G. Salomon, Transfer of learning, in *International Encyclopedia of Education*, 2nd ed. Vol. 2, Oxford, UK: Pergamon Press, pp. 425-441, 1992.

[13] J. Nielsen, “Enhancing the Explanatory Power of Usability Heuristics,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '94)*, Boston: MA, pp. 152-158, April 1994. <https://doi.org/10.1145/191666.191729>

[14] J. Saldaña, *The Coding Manual for Qualitative Researchers*, 3rd ed. Thousand Oaks, CA: SAGE Publications, 2021.

[15] A. M. McAlister, D. M. Lee, K. M. Ehlert, R. L. Kajfez, C. J. Faber, and M. S. Kennedy, “Qualitative Coding: An Approach to Assess Inter-Rater Reliability,” *ASEE Annual Conference & Exposition*, 2017.

[16] S. N. Halpin, “Inter-Coder Agreement in Qualitative Coding: Considerations for Its Use,” *American Journal of Qualitative Research*, Vol. 8, No. 3, pp. 23-43, 2024. <https://doi.org/10.29333/ajqr/14887>



최한길 (Hankil Choi)

2024년~현 재: 서강대학교 가상융합전문대학원 석사과정  
※ 관심분야 : 엔터테인먼트(Entertainment), 3D 그래픽 툴(3D Graphic Tool), UI/UX, 메타버스(Metaverse), 가상융합(Virtual Convergence) 등



최승관 (Seung Kwan Choi)

2021년~현 재: 서강대학교 가상융합전문대학원 엔터테인먼트  
트전공 주임교수 / 전임교수  
※ 관심분야 : 메타버스(Metaverse),  
엔터테인먼트(Entertainment),  
디지털콘텐츠(Digital Contents) 등