

SYSTEM AND METHOD FOR AUTOMATED TRAINING DATA GENERATION FOR AI-ASSISTED DESIGN USING OBJECT GROUPINGS AND SEMANTIC LABELING

0.1 BACKGROUND OF THE INVENTION

0.1.1 Field of the Invention

This invention pertains to the field of computer-aided design (CAD), specifically addressing the generation of training data for AI models used in design applications. The invention resides at the intersection of artificial intelligence (AI), computer-aided design (CAD), computer graphics, machine learning, and design theory.

0.1.2 Background Art

Traditional design processes, particularly in industrial and product design, are time-consuming and require specialized skills. Designers rely on CAD software to create 3D models and engage in iterative rendering processes involving adjustments of materials, textures, lighting, and camera angles to produce visualizations. This iterative process can hinder the exploration of diverse design variations.

Existing AI-based design tools, while offering some automation, often lack the control and precision demanded by professional designers. Many of these tools are based on pre-trained AI models that were trained on generic datasets. These datasets may not capture the nuanced design principles and hierarchical object relationships inherent in specific designs. As a result, AI-generated outputs can deviate from the designer's intent, showing inconsistencies in style or failing to represent the subtle details of the original design.

Prior art exists in related areas:

- US20190156487A1 describes automated generation of pre-labeled training data for machine learning models, focusing on image segmentation and masking for general image processing, not specifically for CAD.
- US20220215145A1 focuses on mesh generation for engineering analysis in computer-aided engineering modeling, not the broader aspects of visual design generation.
- US11308357B2 addresses training data generation for automated driving systems using sensor data, which is distinct from the design-centric data needed for AI-assisted CAD.

Crucially, these existing solutions do not address the unique challenges of AI-assisted design in CAD contexts. They lack the crucial element of **form isolation** through systematic variation. Form isolation is essential for preventing overfitting of AI models on specific stylistic choices present in training data and for preserving the core form intended by the designer. Additionally, existing solutions do not provide the comprehensive suite of features presented in this invention, including advanced user interface controls, location-based background integration, design analysis feedback, and others, as detailed herein.

0.2 SUMMARY OF THE INVENTION

This invention discloses a system and method for the automated generation of training data for AI-assisted design, using object groupings and semantic labeling within CAD models. The invention allows designers to leverage existing CAD functionalities (layers, groups, tags) to organize objects hierarchically, reflecting design principles and relationships. This structure is then used by the system to perform the following:

1. **Systematic Variation:** The system automatically generates a diverse set of variations of the CAD model by systematically altering visual properties (color, texture, material), camera viewpoints (angle, position, focal length), lighting conditions (intensity, direction, color temperature), and backgrounds (color, texture, environment). Critically, this variation is performed while meticulously preserving the core design form as a constant element. This form isolation is crucial for preventing overfitting and enabling the AI model to learn the underlying design intent rather than superficial stylistic features.
2. **Automated Semantic Labeling:** For each generated variation, the system automatically produces descriptive text labels. These labels capture essential attributes of the image, including object groupings, applied variations, and design intent. These labels facilitate the training of the AI model and enable natural language interaction with the system.
3. **AI Model Fine-tuning:** The generated dataset of images and their corresponding semantic labels is used to fine-tune a pre-trained text-to-image AI model, such as Stable Diffusion, using Low-Rank Adaptation (LoRA). LoRA is an efficient technique that allows the model to adapt to the specific design data while preserving its general capabilities learned from a much larger dataset.
4. **AI-Assisted Design Interface:** The invention provides a user-friendly interface for designers to interact with the trained AI model. Designers can input natural language prompts to generate new renderings, specifying adjustments to CMF properties (Color, Material, Finish) and other visual attributes. The interface also includes advanced features:
 - Automated file management
 - Location-based background integration
 - Cost and environmental impact analysis
 - Automated RFQ (Request For Quotation) generation
 - Design callouts
 - Character continuity management
 - Dynamic UI sliders
 - Patent drawing generation
 - Design analysis feedback

- Client annotation tools
- Image security features

0.3 BRIEF DESCRIPTION OF THE DRAWINGS

Figures 1-10 illustrate various aspects of the invention.

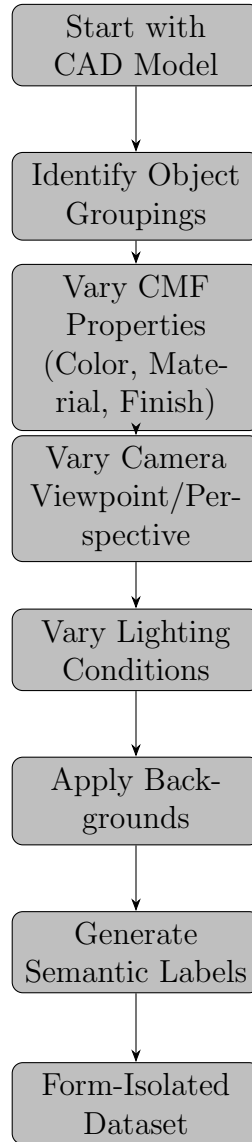


Figure 1: Form Isolation through Systematic Variation Flowchart

- Figure ?? : CAD Model with Object Groupings and Layer Correspondence
- Figure ?? : User Interface for Variation Setup - Selecting the Rendering Mode
- Figure ?? : User Interface for Variation Setup - Entering a Unique Instance Name

- Figure ??: User Interface for Variation Setup - Specifying the Class Name
- Figure ??: User Interface for Variation Setup - Describing the Lighting Setup
- Figure 1: Form Isolation through Systematic Variation Flowchart
- Figure ??: Generating Variations - 'Hero Mode', Product-Style Perspective Renders, and "Orbital Representation"
- Figure ??: Automated Design Callout Generation
- Figure ??: Modular User Interface for Texture and Style Application
- Figure ??: Location-Based Background Integration using Google Maps

0.4 DETAILED DESCRIPTION OF THE INVENTION

0.4.1 Object Grouping in CAD Models

The system is designed to be compatible with various industry-standard CAD software, enhancing its accessibility and integration into existing design workflows. While the principles of the invention apply broadly, specific implementations leverage particular features of different CAD packages. The system requires that the CAD software supports modular object organization and provides programmatic access to model data. These features are crucial for automating variation generation and semantic labeling.

Supported Software Specifically, the current implementation supports the following CAD software:

- **Rhinoceros 3D (Rhino):** Rhino, with its robust scripting API using Python, enables deep integration with the Automated Data Generation Module. Examples of Rhino and Python code provided herein illustrate this integration. Skilled practitioners can readily adapt these examples to other design software and APIs.
- **Fusion 360:** Fusion 360, widely used in engineering and product design, is a target platform for this invention. Its Python API provides functionalities similar to those in Rhino, allowing seamless integration with the Automated Data Generation Module.
- **SolidWorks:** SolidWorks can be integrated with the system through its API, which can be accessed from Python using COM interop, enabling functionalities similar to those in Rhino and Fusion 360. Alternatively, the system can be adapted to directly utilize the native C++ API of SolidWorks.

The core principles of the invention are applicable across different CAD platforms, as long as they offer similar functionalities in terms of object hierarchy and programmatic access.

paragraph Object Grouping Mechanisms

The system is adaptable to different object grouping mechanisms used by various CAD software:

- **Layers (Rhino):** In Rhino, layers (and sublayers) are used for object organization. The provided code illustrates how layer-based groupings correspond to the invention’s object grouping concept. For example, different parts of a chair model are grouped into sublayers like "Chair::Legs," "Chair::Seat," and "Chair::Backrest," allowing for applying material variations based on layer assignment.
- **Components (Fusion 360):** Components in Fusion 360, similar to layers in Rhino, represent distinct parts or sub-assemblies within a design. The system can be adapted to utilize component information for object grouping and variation generation.
- **Groups/Bodies (SolidWorks):** SolidWorks utilizes groups and bodies for organizing geometry. The system can be implemented to recognize these groupings for applying variations and generating semantic labels.

Irrespective of the specific CAD software or its internal representation, the core concept of organizing objects hierarchically to represent design intent remains consistent.

Importing and Preprocessing CAD Data Importing CAD models and any necessary preprocessing steps are essential for compatibility and efficient data handling.

Mesh Conversion In some cases, conversion of CAD geometry into a mesh representation might be necessary for efficient rendering and processing. The system handles mesh data, maintaining object groupings and applying variations to mesh elements. If mesh conversion is required, the system prioritizes using the highest fidelity mesh representation, balancing detail and computational efficiency.

Geometry Simplification For complex CAD models, geometry simplification techniques, such as mesh decimation or NURBS curve reduction, can be used to reduce computational load during variation generation and rendering. The system incorporates parameters to control the simplification level, allowing designers to balance detail with performance. The system provides feedback to the designer if simplification is necessary and allows for adjustment of simplification parameters.

0.4.2 Automated Data Generation Module

The Automated Data Generation Module is the core of the invention, responsible for creating the diverse and richly labeled dataset required to train the AI model. It interacts directly with the CAD system, extracting object groupings, applying systematic variations, and generating corresponding semantic labels.

Identifying Object Groupings: This module analyzes the CAD model to identify object groupings based on user-defined criteria within the enforced hierarchical organization. This module utilizes the Rhino template that enforces the three-level hierarchy (dominant, subdominant, subordinate) and the code traversing layers/sublayers to manipulate objects.

This is illustrated with the example code (`'get_objects_from_sublayers'`) for Rhino, demonstrating the mapping between the CAD model's organization and the system's understanding of design components.

This explicit modularity, provided as layer-based grouping in Rhino (as an example), provides a direct mapping between the CAD model's organization and the system's understanding of design components.

Generating Variations

This module applies systematic variations to the CAD model across four key aspects: visual properties, camera viewpoints, lighting conditions, and backgrounds.

Object Visual Properties

Visual Properties (CMF) The system varies the color, material, and finish (CMF) properties of objects within defined groupings, ensuring a diverse range of variations for the training dataset.

Randomness and Predefined Ranges The system utilizes randomness and predefined ranges for CMF property assignments. The provided code examples demonstrate how a data structure like `'MATERIAL_LIST'` defines a library of materials with associated properties (`name, RGB, ...`).

Camera Viewpoints and Perspectives The system varies the camera perspective to capture the design from different angles and distances, enhancing the AI model's understanding of the form.

Hero Shot and Orbital Representation Two examples implemented in Rhino are 'Hero Shot' and 'Orbital Representation.' 'Hero Shot' clusters camera positions around a designer-chosen focal point, simulating product photography. The `'variation_degrees'` parameter controls the range of camera angles.

Focal Length Variation The `'generate_viewpoint_description'` function shows how focal length variations are implemented.

Lighting The system generates diverse lighting scenarios mimicking various environments and times of day. For example, the function `'generate_lighting_variation'` varies lighting parameters like softness, intensity, and color.

Backgrounds The system applies diverse backgrounds to the rendered images, improving context and visual diversity. It uses a library of predefined backgrounds (colors, gradients, textures, scenes). The system also integrates with Google Maps, allowing real-world locations as backgrounds, with accurate placement and perspective using depth maps and ControlNet.

Semantic Label Generation

For every variation generated, the system creates a detailed semantic label. The ‘generate_{caption}’ function

- Object Groupings and Materials
- Camera Viewpoint
- Lighting Conditions
- Background

Semantic labels are used in file naming or stored in separate data structures (e.g., JSON) linked to the images, ensuring consistency and aiding the AI model’s understanding of the relationship between textual descriptions and visual attributes.

Synthetic Augmentation Techniques The system uses data augmentation to increase dataset diversity and model robustness, including:

- Random Cropping
- Rotation and Flipping (while considering physical interactions and accurate labeling)
- Color Jitter
- Noise Addition

The specific data augmentation techniques are customizable based on the design context and dataset characteristics.

0.4.3 AI Model Training Module

This module imbues the AI model with the ability to understand and generate variations of the designer’s intended form, focusing on the core form independent of stylistic variations.

Pre-trained Model Selection

The system uses a pre-trained text-to-image AI model. Key criteria for selection include:

- Open-Source License: for control over customization, deployment, and future development.
- Architectural Flexibility: adaptability for efficient fine-tuning (e.g., using LoRA).
- Adequate Image Quality: sufficient quality and resolution for design visualization purposes.

Potential candidates include Stable Diffusion and other emerging diffusion-based models known for their flexibility and efficiency.

Training for Form Recognition

The training emphasizes form recognition over stylistic imitation.

Dataset Structure The dataset comprises numerous variations of the same design. The form remains constant while visual attributes (CMF, viewpoints, lighting, backgrounds) are varied, directing the AI’s focus to the consistent form.

Semantic Label Guidance Semantic labels emphasize the underlying form and its constituent parts. They guide the AI to learn a representation of the form decoupled from stylistic variations.

Fine-tuning with LoRA

Low-Rank Adaptation (LoRA) is used to fine-tune the pre-trained model efficiently without full retraining.

Benefits of LoRA

- Targeted Modification: LoRA injects small matrices into specific layers, enabling targeted adjustments without disrupting general capabilities.
- Preventing Overfitting: LoRA’s efficiency reduces the risk of overfitting to specific variations in the form-focused dataset.

Multiple LoRAs can be used in combination, allowing simultaneous rendering of multiple unique objects in a scene.

Evaluation and Validation

The training process is monitored and evaluated to ensure the model’s effectiveness.

- Visual Inspection: Designers assess generated images for fidelity to the original form.
- Quantitative Metrics: Metrics like FID and CLIP Score can be used, but interpreted in the context of form representation rather than photorealism.
- Generalization Tests: Testing on unseen prompts and variations ensures the AI can generalize and generate novel variations while preserving design characteristics.

0.4.4 AI-Assisted Design Interface

This user-friendly interface bridges the designer’s intent and the AI’s rendering capabilities.

Core Rendering Functionality

Natural Language Prompts Designers interact using natural language descriptions for CMF changes, lighting adjustments, viewpoints, and scene elements.

Prompt Parsing and Parameter Translation The interface parses prompts using natural language processing, identifying key elements and translating them into rendering parameters for the AI model. This includes:

- Object References: using object groupings from the CAD model.
- CMF Properties: extracting desired changes to color, material, and finish.
- Lighting and Viewpoint: interpreting descriptions for lighting conditions and camera viewpoints.

Rendering Generation Guided by the extracted parameters, the AI model generates renderings while preserving the integrity of the original form, allowing for manipulation of visual attributes without altering the fundamental design.

Advanced Features

The interface offers a suite of advanced features:

Automated File Management

- Automated Naming: generates descriptive file names based on prompt content.
- Cloud Storage: automatically stores renderings and associated data in the cloud with version control.
- Archiving: facilitates project archiving for easy retrieval.

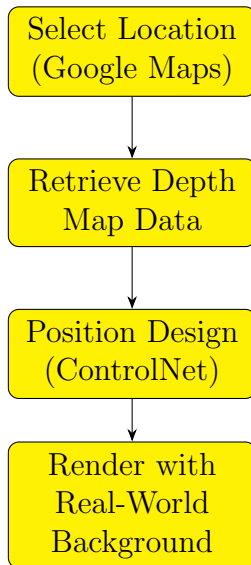
Detailed Close-Ups

- **Region Selection:** Designers can specify regions of interest within the prompt or directly on the rendered image to generate high-resolution close-ups of specific design elements. For instance, a prompt could include "generate a detailed close-up of the joinery details on the armrest." Provided "joinery" was modularized in the pre-training phase – for example, grouped, named, and assigned to the subordinate layer – easy and explicit manipulation is possible
- **Resolution Control:** The system allows for adjustable resolution settings for close-up renders, enabling designers to create highly detailed visualizations for presentations or manufacturing purposes. This can be implemented as a simple slider in the UI, allowing the user to select a desired resolution for the close-up.

Location-Based Backgrounds

- **Google Maps Integration:** The system seamlessly integrates with Google Maps, allowing designers to select real-world locations as backgrounds for their renderings. The user can specify a location and viewpoint directly within the prompt (e.g., "place the bench in Central Park, facing the Bethesda Terrace," as depicted in Figure ??).

- **Depth Map Generation:** The system automatically retrieves depth information from Google Maps data for the chosen location. This depth data is used to create a 3D representation of the environment, which is employed by ControlNet such that the description of the scene elements (central park, New York City Skyline) match the actual contours of the buildings from a particular perspective, which is currently beyond the capabilities of generative AI unaided in this manner. Because the AI understands the essential form of the object and its status, relative to the gravity vector and its class and thus relationship to the world, no manual manipulation of its relationships between the design. This allows the object the background to be rendered within a realistic background context without tedious manipulation of focal length, perspective and object placement. It just looks right.



Mass & Volume Calculations

- **Automated Analysis:** Based on the dimensional information from the CAD model and the CMF properties specified in the prompt, the system automatically calculates the estimated weight, volume, and material costs of the design. For example, if the prompt specifies "brushed stainless steel" for a bench frame, the system will calculate the volume of the frame, reference the density of stainless steel, and then calculate the estimated weight and cost based on current market prices.
- **Environmental Impact:** The system estimates the environmental impact of the design based on the chosen materials and manufacturing processes. This could involve calculating the carbon footprint based on material extraction, processing, and transportation data.
- **Automated RFQ Generation:** The system can automatically generate requests for quotations (RFQs) based on the calculated material requirements and manufacturing specifications. This could involve automatically filling out web forms or generating emails pre-populated with the relevant design and material information, streamlining the process of obtaining quotes from suppliers.

Design Callouts

- **Automated Callout Generation:** generates professional-style callouts on rendered images using image segmentation, highlighting relevant design elements and providing annotations based on prompts or rules.
- **Editable Annotations:** allows designers to edit callout text, positioning, and style.
- **Searchable PDF Export:** embeds callouts as searchable text elements in PDF exports.
- **Consistent Character Appearance:** ensures consistent character appearance across variations.
- **Face Swapping:** employs face-swapping for consistent facial appearance even with variations in other aspects.
- **Text-Based Description Matching:** matches character appearance to designer descriptions from the prompts.

Dynamic UI Sliders

This feature provides an advanced level of interactivity and control within the AI-assisted design interface, allowing designers to fine-tune rendering parameters intuitively.

- **Context-Sensitive Sliders:** The interface dynamically generates sliders for controlling rendering parameters based on the context of the text prompt. For instance, if a designer inputs a prompt like "a bench against a backdrop of a stormy sky and waves," the system recognizes elements such as "stormy sky" and "waves." It then generates sliders corresponding to these elements, such as:
 - **Intensity of the Storm:** Adjusts the severity and visual impact of the stormy sky.
 - **Turbulence of the Waves:** Controls the roughness or calmness of the water.
 - **Brightness of the Lightning:** Modifies the prominence and luminosity of lightning effects.
 - **Time of Day:** Alters the overall lighting and atmosphere, from dawn to dusk.
- **Intuitive Control:** These sliders offer a visual and interactive means for designers to adjust parameters without modifying the text prompt directly. This capability allows for:
 - **Granular Exploration:** Designers can incrementally adjust parameters to observe subtle changes in real-time.
 - **Rapid Experimentation:** Facilitates quick iteration by enabling immediate visual feedback from slider adjustments.

- **Enhanced Creativity:** Encourages designers to explore a wider range of design variations that they might not have considered through text prompts alone.
- **Parameter Mapping:** The system intelligently maps slider values to the AI model’s rendering parameters:
 - **Adaptive Mapping:** Depending on the parameter’s complexity, the mapping can be linear, non-linear, or based on a learned function that correlates slider positions with visual outcomes.
 - **Real-Time Rendering:** Even before the sliders are adjusted, the AI model can cache segmented renders as 2D layers whose opacity or blendings are then easily controlled in real-time or near-real-time, providing immediate visual feedback.
 - **User Customization:** Designers can customize the range and sensitivity of sliders, tailoring the interface to their specific needs and preferences.
- **Integration with Other Features:** The dynamic sliders work in conjunction with other interface elements:
 - **Preset Management:** Slider configurations can be saved as presets for future use or sharing with team members.
 - **Modular UI:** Sliders are part of the modular user interface, allowing designers to add or remove them based on the design context.
 - **Design Analysis Feedback:** Adjustments made via sliders can be analyzed by the system to provide feedback or suggestions, enhancing the iterative design process.

Illustrative Example:

Consider a scenario where a designer is creating a rendering of a modern living room featuring their custom-designed chair. The prompt might be: "A modern living room with large windows overlooking a cityscape, featuring my custom-designed chair in the center."

Based on this prompt, the system generates sliders for:

- **Window Size:** Adjusts the size of windows in the scene.
- **Cityscape vibe intensity:** Controls visual tension and presence inherent with visualized human activity in view, modifying for example the number of cars on the street, the density of pedestrian foot traffic, the city view (e.g., daytime vs. nighttime).
- **Ambient Light vs Exterior light:** Modifies the balance of interior lighting vs the intensity of light sources outside the room.

By adjusting these sliders, the designer can fine-tune the rendering to achieve the exact visual effect they desire, such as transitioning from a bright morning scene to a cozy evening setting without changing the text prompt.

Annotation Placement: Annotations such as dimension lines, reference numerals, and technical labels are automatically placed according to patent drawing conventions. The system uses the structured data from the CAD model to accurately determine dimensions and relationships between design elements, ensuring that the annotations are placed correctly and comply with patent regulations.

Design Analysis

- **Feedback on enviornmental context:** The system integrates with a large language model (LLM) like GPT-r or a fine tuned or specialized LLM. The CMF choices of the designer and the enviornment are combined with existing logic and natural language processing to produce a prompt that asks a LLM to interrogate potential issues that arrise from the new context of the object in its pictured enviornment. For example, if the prompt specifies "a park bench made of unfinished hickory wood in West Virginia," the LLM could access external knowledge bases to determine that hickory is susceptible to carpenter ant infestations in that region, flagging a potential durability issue. A similar issue could be raised around material suitability with regards to corrosion if a designer specified an enamel painted low carbon steel for a seaside bench near salt spray. This feedback can be presented as text-based warnings, questions. Integration options for commercial suppliers of coatings and finishing are present, for example, recommending a durable two-part epoxy coating for a part that will be exposed to electrolytic solvents such as salt water, or adertised a new clear coat if the designer specifies "ultra high gloss" in their natural language CMF prompt. For vendors with products integrated into the system, visual highlights on the rendered image, or even suggested modifications to the design or materials. Loras can be created for vendors to enter their coatings into the system (real world pictures modifying form and enviornments exentisvely but keeping the coating the same.) This allows us to all but promise that the industrial deisgn object, once pained with 'rustolium red' will look like the picture of 'rustolium red'
- **Contextual Feedback:** The design analysis takes into account the specific context described in the prompt. For example, if the prompt mentions "outdoor furniture," the LLM might check for weather resistance and UV degradation of the chosen materials. The more context the designer provides in the prompt, the more refined and relevant the design analysis feedback will be.

Client Annotation Tools

- **Collaborative Review:** The interface provides tools for clients and reviewers to add annotations, comments, and feedback directly on the rendered images, as illustrated in Figure ???. This facilitates a more streamlined and collaborative review process, eliminating the need for separate feedback documents or email chains.
- **Visual Feedback:** Clients can draw directly on the image, highlight specific areas, add text boxes, or use pre-defined stamps (e.g., "approve," "revise"). This visual feedback

is more intuitive and efficient than text-based comments alone, allowing clients to clearly communicate their thoughts and preferences.

- **Revision Tracking:** The system tracks all annotations and comments, creating a clear record of the feedback and facilitating iterative revisions. Designers can easily see which areas have received feedback and address specific concerns raised by the client.

Image Security

- **Watermarking:** Invisible watermarks are embedded within generated images to protect the designer's intellectual property. These watermarks can contain information about the designer, the project, or the date of creation and can be used to track the origin of the image if it is distributed without authorization.
- **Serialization:** Each rendered image is assigned a unique serial number that can be used to track its usage and distribution. This serial number can be linked to specific client information, allowing the designer to monitor who has access to which images.
- **Usage Monitoring:** The system can detect if a watermarked image is used without authorization or if it appears in unauthorized locations online. This monitoring could be automated using image recognition techniques and web crawling tools, providing alerts to the designer if unauthorized use is detected.

Vision Tracking & Form Review

- **Eye-Tracking Integration:** The system can optionally integrate with eye-tracking technology, using the embedded webcam on a laptop or the rear-facing camera on a mobile phone, for example, to analyze user gaze patterns during the review process.
- **Heatmap Visualization:** Eye-tracking data is visualized as heatmaps overlaid on the rendered images for the designer to review. The heatmap can indicate which parts of the design are calling the viewer's attention. The degree to which the viewer's attention aligns with the intended hierarchy of design elements can be quantified, providing valuable insights to the designer. It allows for an objective assessment of whether the viewer is noticing the intended focal points of the design or if their attention is drawn to unintended areas. This feedback can guide the designer in refining the composition, proportions, repetition, and contrast of the design elements to achieve the desired visual impact.

Modular UI

- **Drag-and-Drop Interface:** Designers can drag and drop images or textures onto the design to apply materials or patterns, creating a more intuitive and direct way of customizing the rendering. For example, a designer could drag an image of a specific wood grain onto the bench seat to apply that texture, or drag a color swatch from a palette onto the legs to change their color.

- **Texture Blending:** The system supports blending multiple textures together to create more complex and nuanced material effects. For instance, a designer could blend a rough stone texture with a polished metal overlay to create a unique surface finish, allowing for a greater degree of creative control over the material appearance.

Style References

- **Style Transfer:** Designers can apply different artistic styles to renderings by providing reference images. For example, dragging and dropping a still frame from a Pixar movie could instruct the system to render the design in a similar style, even if the prompt describes realistic materials. This allows designers to explore a wide range of aesthetics beyond photorealism.
- **Dynamic Style Parameters:** Sliders control the strength and intensity of the applied style. This allows for fine-grained control over the style transfer effect, enabling the designer to blend the reference style with the original rendering to achieve the desired look.

Thematic Sliders

- **Mood and Atmosphere:** Sliders control the overall mood and atmosphere of the rendering. For example, sliders for "brightness," "contrast," "saturation," "warmth," "sharpness," and "depth of field" allow the designer to create a specific emotional tone or visual style without needing to describe these qualities in the text prompt.
- **Style Presets:** Thematic sliders can be combined into presets, allowing designers to quickly apply predefined stylistic looks to their renderings (e.g., "cinematic," "retro," "minimalist").

Preset Management

- **Saving and Loading Presets:** Designers can save their preferred rendering configurations as presets, including CMF choices, lighting settings, camera viewpoints, style references, and thematic slider settings. This allows for the reuse of successful rendering styles across different projects or design variations.
- **Sharing Presets:** Presets can be shared among team members or publicly, fostering consistency and collaboration in design visualization. This can be particularly useful for maintaining a consistent brand identity or visual style across a design team.

Export Options

- **Image Formats:** The system supports exporting renders in various image formats, including JPG, PNG, TIFF, and EXR, catering to different resolution and quality requirements. This flexibility ensures compatibility with a wide range of design workflows and presentation formats.

- **3D Printable Models:** Designers can export the design as a 3D printable model in formats like STL or OBJ, allowing for rapid prototyping and physical evaluation of the design.
- **Foldable Paper Models:** The system can generate patterns for creating foldable paper models of the design, providing a tangible and interactive way to explore the form. This can be especially useful in the early stages of design development or for educational purposes.
- **Vector Graphics:** Designers can export the rendering as vector graphics (e.g., SVG), enabling scalability and integration with other design software. Vector graphics are particularly useful for creating illustrations, logos, and other design assets that need to be resized without losing quality.

History Tree and Visual Diff

- **Iteration Tracking:** The interface maintains a history of all design iterations, including changes to the prompt, applied materials, lighting settings, and camera viewpoints. This provides a clear record of the design process and allows designers to easily revert to previous versions if needed. This history can be visualized as a tree structure or timeline, providing a clear overview of the design’s evolution.
- **Visual Diff:** The system highlights the differences between two selected versions of the rendering, visually showing the changes that have been made. This can be implemented by overlaying the two versions and highlighting the areas that have changed, making it easy for the designer to identify the impact of specific modifications. This feature facilitates a clear understanding of the design evolution and aids in iterative refinement.

This AI-Assisted Design Interface, with its core rendering functionality and extensive suite of advanced features, empowers designers to leverage the power of AI for creative exploration, efficient workflow, and effective communication. By focusing on natural language interaction, form preservation, and a user-centered design, the interface creates a seamless and intuitive experience that enhances the designer’s creative process and unlocks new possibilities in design visualization. The integrated file management, analysis tools, collaborative features, and security measures further streamline the design workflow and address the practical needs of professional designers.

0.5 CLAIMS

1. A computer-implemented system for generating training data for an AI-assisted design system, comprising:
 - (a) a processor;
 - (b) a memory coupled to the processor and storing instructions that, when executed by the processor, cause the system to perform operations comprising:

- i. receiving, at a CAD interface, a CAD model having a plurality of objects grouped according to a hierarchical structure;
 - ii. accessing, using an automated data generation module, the hierarchical structure of the CAD model;
 - iii. systematically varying, using the automated data generation module, at least one of: visual properties of the objects in the CAD model, viewpoints, lighting conditions, and backgrounds while maintaining a core design form of the CAD model, thereby generating a plurality of variations of the CAD model;
 - iv. generating, using the automated data generation module, a semantic label for each variation of the CAD model, the semantic label including information about object groupings, applied variations, and design intent; and
 - v. training, using an AI model training module, a text-to-image AI model using the plurality of variations and corresponding semantic labels.
- 2. The system of claim 1, wherein the visual properties varied by the automated data generation module include at least one of: color, material, finish, texture, camera viewpoint, and lighting condition.
- 3. The system of claim 1, wherein the automated data generation module varies camera viewpoints using at least one of: hero shot positioning and orbital representation.
- 4. The system of claim 1, wherein the AI model training module utilizes Low-Rank Adaptation (LoRA) to fine-tune a pre-trained text-to-image AI model.
- 5. The system of claim 1, further comprising an AI-assisted design interface configured to:
 - (a) receive natural language prompts from a user;
 - (b) generate renderings of the CAD model based on the natural language prompts and the trained AI model;
 - (c) allow the user to adjust visual attributes of the CAD model without altering the core design form.
- 6. The system of claim 5, wherein the AI-assisted design interface further comprises at least one of: automated file management, location-based background integration, cost and environmental impact analysis, automated RFQ generation, design callouts, character continuity management, dynamic UI sliders, patent drawing generation, design analysis feedback, client annotation tools, and image security features.
- 7. A computer-implemented method for generating training data for an AI-assisted design system, comprising:
 - (a) receiving, at a CAD interface, a CAD model having a plurality of objects grouped according to a hierarchical structure;
 - (b) accessing the hierarchical structure of the CAD model;

- (c) systematically varying, using at least one processor, at least one of: visual properties of the objects in the CAD model, viewpoints, lighting conditions, and backgrounds while maintaining a core design form of the CAD model, thereby generating a plurality of variations of the CAD model;
 - (d) generating a semantic label for each variation of the CAD model, the semantic label including information about object groupings, applied variations, and design intent;
 - (e) training a text-to-image AI model using the plurality of variations and corresponding semantic labels.
8. A non-transitory computer-readable storage medium storing instructions that when executed by at least one processor cause the at least one processor to perform the steps of claim 7.

0.6 CONCLUSION

This invention significantly advances the field of AI-assisted design by addressing key challenges in training data generation, workflow integration, and designer control. By focusing on form isolation and incorporating a comprehensive suite of advanced features, the system empowers designers to leverage the creative potential of AI while maintaining full control over their design intent and streamlining their workflow. The provided code examples offer a glimpse into the technical implementation of the system, demonstrating how the core functionalities are realized in practice.