DRAWING AUTOMATION APPROACH USING SOLIDWORKS API

Andrea Gutai [1] [ORCID 0000-0002-3315-4108], *Valentina Dukić* [1] [ORCID 0000-0001-7733-1236], *Andraš Anderla* [1] [ORCID 0000-0003-4642-9382], *Srđan Sladojević*[1] [ORCID 0000-0002-9533-1210], *Milan Mirković* [1] [ORCID 0000-0002-9128-4453]

[1] *Faculty of Technical Sciences, Department of Industrial Systems and Management, Novi Sad*

Abstract: *The automation of repetitive processes included in the design phase represents an emerging trend in the CAD industry. Automation plays a crucial role in the reduction and minimization of time-consuming and error-prone tasks. In technical drawings it provides a higher level of precision by ensuring that standards and design principles are applied uniformly across all drawings. The aim of the paper is to improve the drawing process in terms of modification efficiency and quality, leading to a shortened time required to complete a design and accelerate time to market. In this paper, the automation process for creating drawings is proposed using SolidWorks API macros. In order to create standardized documentation, a macro is created in which tasks related to views, dimensions, formats, and annotations are unified and automated. With the suggested technique, new drawings can be made with minimal user input based on the CAD model.*

Key words: SolidWorks, API, macro, CAD drawing

1. INTRODUCTION

Organizations are progressively moving towards mass customization and design automation in order to remain competitive in the market, fulfill customer requirements, and retain design quality. In a recent whitepaper by SolidWorks (Dassault Systèmes, 2022), it is stated that according to studies, 63% of engineers have encountered demands for customization, and 26% expect a growing rate in the upcoming years. A tremendous increase in recent years has raised new research questions related to faster design while simultaneously maintaining the same level of quality. Design automation is one of the approaches used for the elimination of errors and shortening time by extracting information from the knowledge base comprising previous experiences, standards, and logical relations between variables. Along with the development of computer-aided design (CAD) systems, it is feasible to develop macros, applets and software tools that can support a variety of engineering processes, including the design and manufacture of machine components (Tzotzis et al., 2020).

Configurations represent a low level of design automation and are mostly used with larger groups of similar products. Dimensions can be easily stored in only one file with variations in dimensions, eliminating the need for numerous models. This approach is applicable to the creation of parts, but drawing generation remains an area of discussion in research and academia.

Detailed technical drawings convey the design purpose of the object that needs to be manufactured and assembled. Additionally, another benefit to the manufacturer is the transmission of a plethora of information related to the required specification, form, and function. Clear and precise instructions are required to guarantee that the finished product meets predefined requirements, but generating drawings brings a lot of challenges if the person is not proficient and knowledgeable enough. On the other hand, the amount of time that is required represents another drawback of manual drawing creation.

Numerous CAD softwares are used in the creation of three-dimensional (3D) models and technical drawings, and consequently, a two-dimensional (2D) technical drawing can be generated based on a 3D model. Technical drawings include orthogonal projections and views, annotations, dimensions, and different types of lines. To achieve automation of the aforementioned process and address this challenge, a Solidworks macro is proposed. The approach is tested on several drawings and compared against the manual drawing process and its length.

The remainder of the paper is structured as follows: Section 2 represents related field work, while Section 3 represents a methodology for drawing automation utilizing the SolidWorks Application Programming Interface (API). Section 4 depicts a discussion of the results obtained and reported in Section 3. Finally, Section 5 discusses the study's conclusions and limits, as well as future research objectives.

2. RELATED WORK

In order to reduce time needed for the human design process, design automation is proposed as a set of predefined codes used to automatically complete operations. Crucial step is strictly adhering to a set of predetermined procedures that should be identified based on knowledge base and previous experience. The design automation process consists of four steps that are recurring and identified by Sirigiri et al(Sirigiri & Esanakula, 2022). The first step is related to reviewing the whole design process, followed by the recognition of repetitive tasks that should be taken into consideration for automation. Subsequently, the development of rule-based design steps for custom features or components is the next phase and, in the end, implementation is performed using SolidWorks API functionalities.

In Figure 1, a schema is presented for a knowledge-based system for CAD modeling proposed by (Jayakiran Reddy et al., 2018). Every system must be linked to a knowledge base that comprises experience and knowledge rules applied by design professionals. Domain expertise and standards are used as support in building a knowledge base. Besides international standards, each organization has its own internal standardization and rules for drawings that should be taken into consideration when developing a system. Domain specialists are individuals who perform these tasks manually on a daily basis and can define the business logic that will be implemented. A domain expert has a goal to find the optimal solution among the standards to accurately represent the design intent, which is the functionality of the product being represented. This goal can be attained if the dimensions are defined according to predefined guidelines to ensure that the original objectives are met and cannot be derived from the geometry of the model. The knowledge base is stored in such a way that a program can retrieve the data whenever the system necessitates it without human interaction. The user has only access to a graphical user interface that is connected to the inference engine, which employs logical rules and regulations to find a solution to the query submitted by the user.

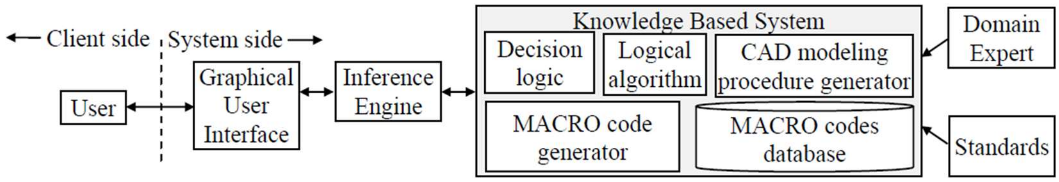


Figure 1: Schema for knowledge-based system for CAD modeling (Jayakiran Reddy et al., 2018)

The illustrations on engineering drawings are based on orthogonal projection views and dimensions and annotations. In engineering drawings, dimensions are frequently used to indicate the measurements and positions of specific geometric components. Research related to the automation of the drafting process is mostly focused on automatic dimensioning.

Focus on the elimination of redundant or insufficient dimensions was proposed in approach by (Sui et al., 2013). All dimensions on a technical drawing are changed to a non-directed graph, and traversing adjacent non-directed graph matrices is used for the determination of redundant or insufficient dimensions.

Li et al. proposed an automatic approach for dimensioning engineering drawings of mechanical products based on usage of the curve chain (Li & Yang, 2020). A geometric topological structure between elements is established. Dimensions within the chain are classified as absolute dimensions in the curve chains generated above, whereas dimensions between chains are classified as relative dimensions. Results have shown that redundancy is avoided with this approach.

Raffaeli et al. proposed an approach for automatic generation of technical drawings for gear motors product family and for testing technical drawings with the aid of feature recognition algorithms (Raffaeli et al., 2016) A scheme is corresponding to variants of the same product family and consists of rules related to views, dimensions and positions of them on drawing, and it is entered in the off-line phase by a designer using a dedicated graphical interface. The second phase relates to repetitive and autonomous use of the system to develop dimensioned drawings.

Not only does the mechanical industry involve product design automation, but also methodology for furniture design, specifically coffee tables, is proposed by (Manavis et al., 2022). Two case studies were proposed, and one of them is based on SolidWorks APIs that are based on Visual Basic for Applications (VBA). Estimated time with the traditional way of design is approximately 85 minutes, and with SolidWorks VBA application it required 3.4 seconds. Indisputably, design automation reduces the time required for manual processes with implementation in numerous industries.

3. METHODOLOGY

This section provides an in-depth description of the SolidWorks API, covering its key components and functions. It explains the programming languages supported by the API, the accessible libraries and resources, and the integration options with external applications and data sources. The report emphasizes the flexibility and extensibility of the proposed SolidWorks API macro, which allows researchers to customize the methodology to their own requirements.

Proposed methodology is based on the SolidWorks API, a popular CAD product used by many industries with the usage of VBA. It represents a programming language that is used for automation of tasks by writing macros. Visual Basic is an object-oriented programming (OOP) language that necessitates knowledge and understanding of object-oriented design and how objects are formed and implemented (Zeid, 2015).

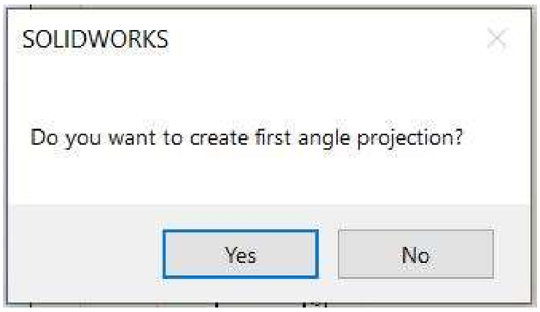
The SldWorks interface is the highest-level object in the SolidWorks API stack. All other interfaces exposed in the SolidWorks API object hierarchy are accessible via SldWorks and IModelDoc2 interface is located just beneath the SldWorks interface and provides access to the three core Solidworks document types: parts, assemblies, and drawings (Buric et al., 2021). Each document type has its own object IPartDoc, IAssemblyDoc, and IDrawingDoc and a set of related functions. IDrawingDoc is used for proposed methodology in view of the fact that it provides access to functions that execute drawing operations in SolidWorks.

Component Object Model (COM) automation is used to interact with SolidWorks software programmatically with scripts in programming (Carlberg et al., 2016). A user interface action calls a programmed function in the programming tool, which then triggers a defined operation in the CAD system that should be proceeded based on selected features. Due to its relationship with VBA, which is an object-oriented SolidWorks API function with objects, methods are used for performing operations on objects, while properties represent features or attributes of specific objects. In Table 1, all methods and properties that are used in methodology are shown with additional details for each method or property in terms of functionality and return values. Due to the version of SolidWorks, the names of the methods or properties can be differently formed compared to the ones shown in Table 1.

Table 1: API methods and properties used in methodology

API Method/Property	Explanation of functionality	Return/Property value
ActiveDoc	Connects to currently active document	Model document
NewDocument	Creates a new document with specific template	Newly created document
GetConfigurationNames	Get the names of the configurations in this document.	Array of the names of the configurations in the order they are created in document
Create1stAngleViews2	Generates three typical orthographic views (first angle projection)	True/false
Create3rdAngleViews2	Generates three typical orthographic views (third angle projection)	True/false
GetPathName	Gets the full path name for this document, including the file name	Full path name for this document including the file name as string
CreateDrawViewFromModelView3	Creates a drawing view on the current drawing sheet using the specified model view	Pointer to the newly create IView object (drawing views found in a drawing document)
ViewDisplayShaded	Sets the current display mode to Shaded	None
InsertModelAnnotations3	Inserts model annotations into the currently selected drawing view of this drawing document	Array of inserted IAnnotation objects
InsertCenterMark3	Inserts a center mark in a drawing document.	Center mark
SaveAs	Saves current document to specified format and name	True/false

The flow consists of steps related to activating the Solidworks program, opening a part, and making a new drawing for an opened model with a specified template. Firstly, variables related to the SolidWorks application, model, drawing, view, dimension, and annotation objects are defined. Initialization of the swApp variable with the current SolidWorks application and the swModel variable with the active document. The following step is binding to SldWorks and connects to the active model, and if the model is not opened, the user also has an option to open the model. It checks whether there is an active document open in SolidWorks, and if not, a message box is displayed to handle errors. A new drawing document is created using the NewDocument method with the specified template name, paper size, and scaling factors. To provide a user interface (UI) for a SolidWorks API macro, users can use forms or dialog boxes to allow users to input parameters, select different options, and interact with the macro. For the purpose of methodology, a simple VBA message box (MsgBox) is created, and it accepts arguments and receives simple user input to choose whether first or third angle projections should be used in a specific drawing. In Figure 2, the message box is depicted.


SOLIDWORKS

Do you want to create first angle projection?

Figure 2: MsgBox in the beginning of the process

The outcome of the function is determined by the button that the user clicks, in this case Yes or No. If the user selects Yes, the first angle drawing is created, while No, conversely, will create the third angle drawing. Based on the user response, either the Create3rdAngleViews2 or Create1stAngleViews methods are called to create the appropriate views in the drawing.

The following step involves placing standard views into the sheet format as well as adding isometric view, annotations, and dimensions. The CreateDrawViewFromModelView3 method is used to create an isometric view in the drawing. In each case (First or Third Angle), specific positioning values of X and Y coordinates of the isometric view are adjusted to the position of other views. The ViewDisplayShaded method is called to display the isometric view with a shaded appearance. The InsertModelAnnotations method is called to insert model annotations (dimensions and notes) in the drawing, while the InsertCenterMark3 method is used to insert a center mark at a required position in the drawing.

Finally, the last step is related to saving drawing documents and exiting the Solidworks application are the final steps with the SaveAs method. Each drawing at the end of the process is automatically saved in DWG and PDF formats in a predefined location.

4. RESULTS AND DISCUSSION

In this section, results of the proposed methodology for automating the drawing process utilizing a SolidWorks API macro are presented. Simple 3D model is used for testing purposes, and following that, the model is opened, and standard views based on the user's preference are generated. Users had a chance to open their part and to choose through message box whether it will be drawn in the first or third angle. After selection, three default views are generated in both cases but in different positions on the paper, including also isometric view in shaded display mode. Annotations, dimensions, centerlines and centermarks are automatically applied in both situations and Figure 3 reflects the outcome of the proposed macro.

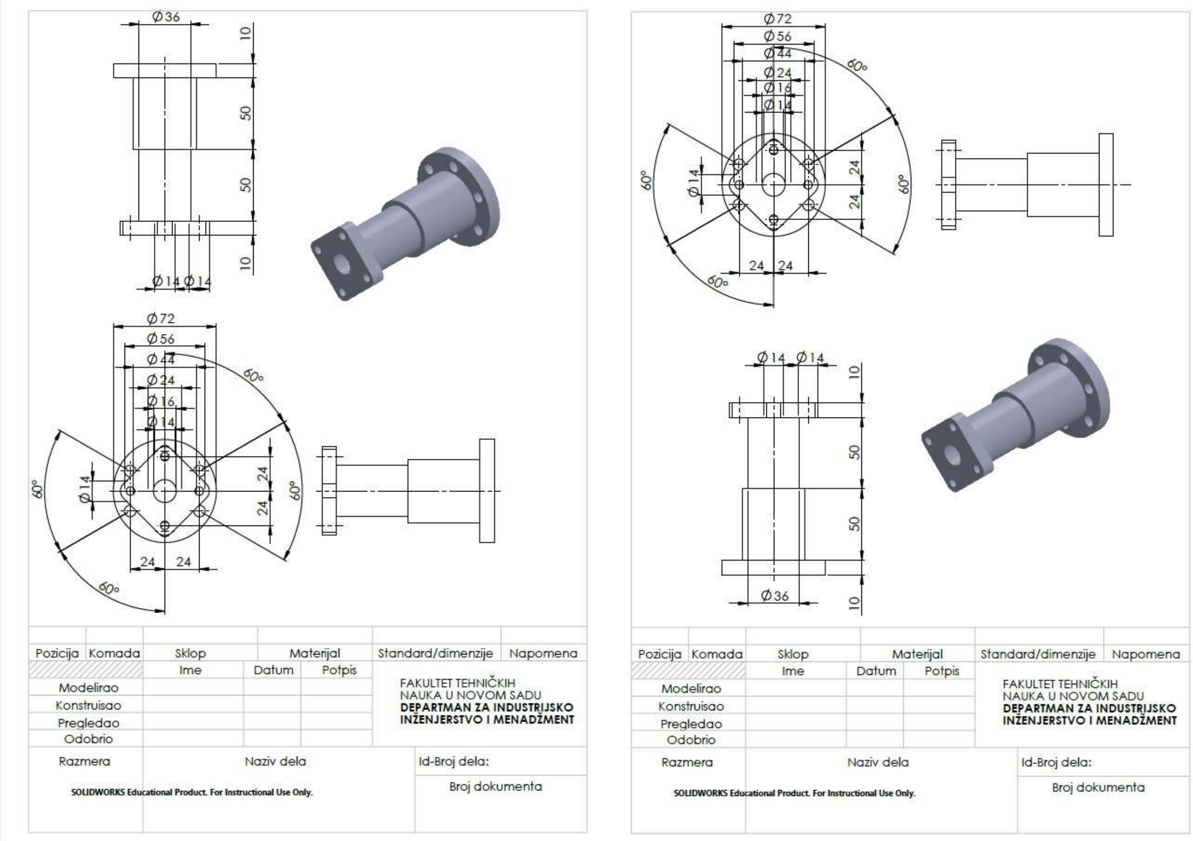


Figure 3: First Angle (left) and Third Angle drawings (right)

Nevertheless, methodology demonstrated great results in terms of saving time when talking about creation of drawings. Manually generating the first or third angle a drawing and saving lasted around 81 second, but executing the macro only required 6 seconds. Compared to manual time for creating or drawing with proposed macro time is reduced by 92.59%. Even if minor adjustments are required in order to make corrections to automatically created drawing, it is unquestionably faster compared to the entire manual process. The knowledge base can be expanded with various rules related to drawing generation, views and annotations and thus contribute to the improvement of the current version of macro.

5. CONCLUSIONS

The aim of the paper is to present the development of drawing automation in SolidWorks API using VBA. The primary goal of this work is to reduce time and inaccuracies in engineering drawings. In comparison to manually created drawings, it includes minor variations that can be easily addressed manually by an engineer or design expert at the end of the process. Automation of designing tasks can minimize conventional and tedious processes.

As a result, it's critical to keep everything simple and widely understandable. It is tested on a smaller group of similar products so future research will be in terms of testing it with a larger set of models. For more precise findings, a larger group of models should be evaluated, but the creation of macro should be centered on a similar set of products, so it should still be adapted to the needs of a specific group of products. Future research will include, in addition to parts, assemblies that will have Bill of Material (BOM) and numerous sheets to collect all drawings related to all parts in assembly.

REFERENCES

Buric, M., Brcic, M., & Skec, S. (2021). Towards Automated Drafting in CAD Systems. *ACM International Conference Proceeding Series*, 233–238. https://doi.org/10.1145/3508297.3508335

Carlberg, G., Söderby, E., Blomqvist, S., & Amadori, K. (2016). *Context-based 3D CAD configuration using object-oriented programming-A case study within the construction industry*. www.liu.se

Dassault Systèmes. (2022). *Using Design Automation to Reduce Costs, Increase Profitability*. SolidWorks. https://www.solidworks.com/media/white-paper-using-design-automation-reduce-costs-increase-profitability

Jayakiran Reddy, E., Venkatachalapathi, N., & Pandu Rangadu, V. (2018). Development of an approach for Knowledge-Based System for CAD modelling. *Materials Today: Proceedings*, *5*(5), 13375–13382. https://doi.org/10.1016/j.matpr.2018.02.330

Li, G., & Yang, P. (2020). Automated dimensioning method of engineering drawings for mechanical products based on curve chain. *Computers, Materials and Continua*, *65*(1), 847–863. https://doi.org/10.32604/cmc.2020.011398

Manavis, A., Tzotzis, A., Tsagaris, A., & Kyratsis, P. (2022). A Novel Computational-Based Visual Brand Identity (CbVBI) Product Design Methodology. *Machines*, *10*(11). https://doi.org/10.3390/machines10111065

Raffaeli, R., Cicconi, P., & Germani, M. (2016). Automation of drafting execution by schemes definitions and feature recognition. *Computer-Aided Design and Applications*, *13*(4), 459–470. https://doi.org/10.1080/16864360.2015.1131538

Sirigiri, B. C., & Esanakula, J. R. (2022). A Basic Automated CAD Modelling Approach for an IC Engine Piston. *AIP Conference Proceedings*, *2648*. https://doi.org/10.1063/5.0114356

Sui, T. Z., Tan, Z., Wang, L., Gu, X. Bin, & Ren, Z. H. (2013). A method for completeness testing of dimensioning in 2D drawing. *Applied Mechanics and Materials*, *319*, 351–355. https://doi.org/10.4028/www.scientific.net/AMM.319.351

Tzotzis, A., García-Hernández, C., Huertas-Talón, J. L., & Kyratsis, P. (2020). Cad-based automated design of fea-ready cutting tools. *Journal of Manufacturing and Materials Processing*, *4*(4). https://doi.org/10.3390/jmmp4040104

Zeid, I. (2015). *Mastering SolidWorks: The Design Approach*. Pearson.