Reslice3Dto2D: Introduction of a software tool to reformat 3D volumes into reference 2D slices in cardiovascular magnetic resonance imaging

Darian Viezzer[1,2,3,*], Maximilian Fenski[1,2,4], Thomas Hiroshi Grandy[1,2,4], Johanna Kuhnt[1,2], Thomas Hadler[1,2,3], Steffen Lange[5], Jeanette Schulz-Menger[1,2,3,4]

[1] Charité – Universitätsmedizin Berlin, corporate member of Freie Universität Berlin and Humboldt-Universität zu Berlin, ECRC Experimental and Clinical Research Center, Lindenberger Weg 80, 13125 Berlin, Germany

[2] Working Group on Cardiovascular Magnetic Resonance, Experimental and Clinical Research Center, a joint cooperation between the Charité – Universitätsmedizin Berlin and the Max-Delbrück-Center for Molecular Medicine, Berlin, Germany

[3] DZHK (German Centre for Cardiovascular Research), partner site Berlin, Berlin, Germany

[4] Helios Hospital Berlin-Buch, Department of Cardiology and Nephrology, Berlin, Germany

[5] Hochschule Darmstadt (University of Applied Sciences), Faculty for Computer Sciences, Darmstadt, Germany

* Corresponding author, E-Mail: darian-steven.viezzer@charite.de

User Manual

Software: Reslice3Dto2D
Version: 3.0
Project Homepage: https://github.com/DSV-CUB/Reslice3Dto2D
License: Copyright   (c)   2024   Darian   Steven   Viezzer,
                    Charité – Universitätsmedizin Berlin

Introduction

This is the User Manual for the Reslice3Dto2D software that enables the user to reformat 3D data into 2D slices at the exact same location as a reference 2D acquisition including options to specify the slice thickness and slice profile.

Installation

The Reslice3Dto2D is implemented in Python (Version 3.8, Python Software Foundation). Accordingly, Python needs to be installed in the specified version from https://www.python.org if it is not yet installed on the machine. For Microsoft Windows ensure that during installation Python is added to the system environment variable PATH. The Reslice3Dto2D can be accessed via the publication by Viezzer et al. or the GitHub project page under https://github.com/DSV-CUB/Reslice3Dto2D. After download of the source code, unpack the compressed file. For Microsoft Windows please run via double click the INSTALL_windows.bat to install all necessary Python site packages. After successful installation, the tool can be started via RUN_windows.bat. For macOS users, please open a Terminal and go to the directory of the Reslice3Dto2D tool. Run the following two commands, to make the command files executable:

chmod 700 INSTALL_macOS.command
chmod 700 RUN_macOS.command

You can now close the Terminal and install the necessary site packages via double click on INSTALL_macOS.command. After successful installation, the RUN_macOS.command starts

the application. In the current version, the following site-packages of Python are necessary: numpy, pydicom, PyQt5, qimage2ndarray and scipy.

Usage

The Reslice3Dto2D tool includes a small graphical user interface (GUI) to facilitate the usage as shown in Figure 1 and described in detail in the following.
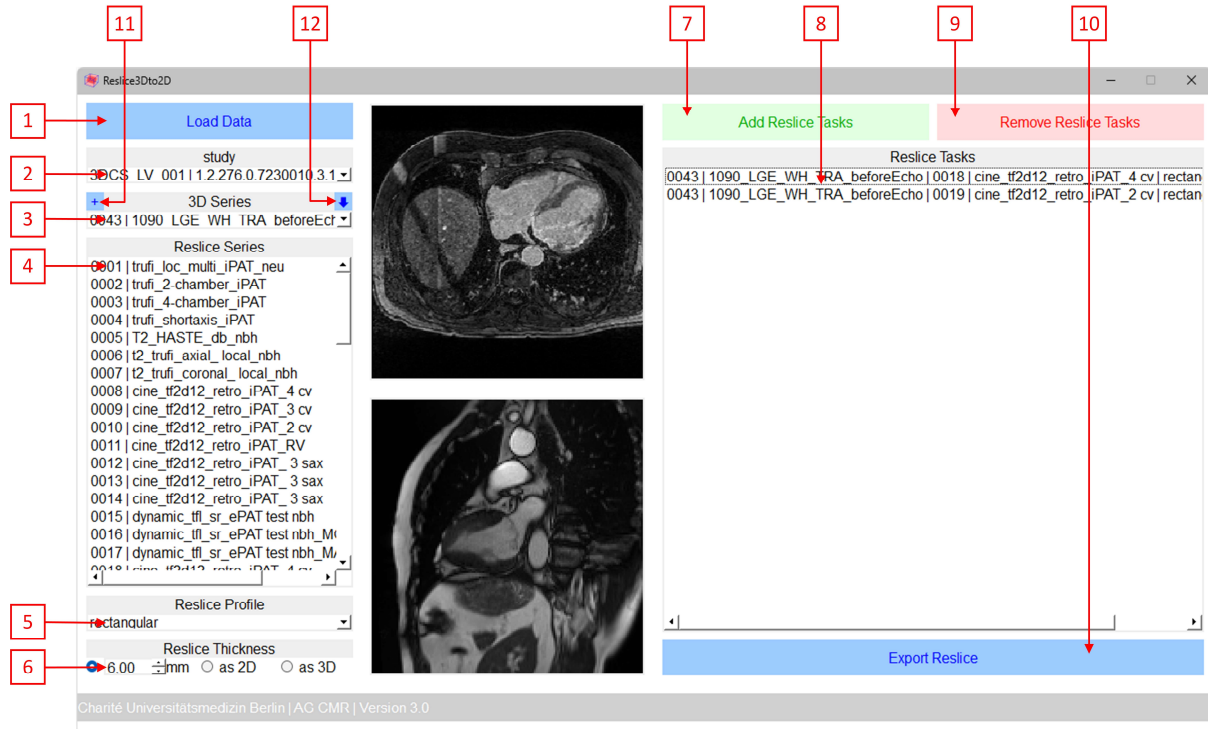

Figure 1: Reslice3Dto2D graphical user interface (GUI)

The Load Data button [1] opens a dialog to choose a directory with DICOM data that is accordingly loaded. If the loaded data includes multiple examination, then the study selection [2] appears, otherwise it is hidden. The study selection enables the user to switch between the studies and provides the patient's name as well as the study UID. The tool automatically detects 3D data and lists them in a drop-down menu [3]. The reslice serieses list [4] shows all serieses of an examination. Any series can be used as the reference 2D series. Additionally, the tool offers the user to select the slice profile [5] and the slice thickness [6]. The latter can be explicitly set, read from the reference slices (as 2D) or read from the selected 3D data (as 3D). As 3D data is stored in a stack of slices, the slice thickness is the distance between those slices and not the in-plane pixel spacing. After choosing the 3D series, all reference 2D slices and the accordingly profile, the reformatting tasks can be added [7] to the task list [8]. If multiple 3D series should be reformatted, all tasks can be added. Individual tasks can be excluded [9]. If all tasks are collected, then the reformatting is performable and exportable [10]. The tool exports the reformatted data in DICOM format by using the DICOM tag information from the original 2D slice and overwriting the series number, series description, protocol name, trigger time, instance number, series instance UID, SOP instance UID and media storage SOP instance UID. It is important to load the complete case into the tool, as otherwise the generated series number collides with an already assigned one. The series description and protocol name include R3D2D at the beginning to mark that it was reformatted. The Reslice3Dto2D tool is aware of the slice location and phase. Therefore, the reformat of a 3D dataset according to a 2D cine will result in a single image instead of a multitude of the same image as the location during a heart cycle is the same. Furthermore,

2

time resolved 3D data (like 3D CINE or 4D Flow) can be reformatted as well resulting in an according 2D CINE or 2D Flow data. The reformatting of 3D data according to a reference 3D data is also possible, as the underlying 2D DICOM slices of the reference 3D data are used.

Additionally, the Reslice3Dto2D enables to stack data to a 3D dataset [11], which opens a dialog as shown in Figure 2, and export the 3D data [12].
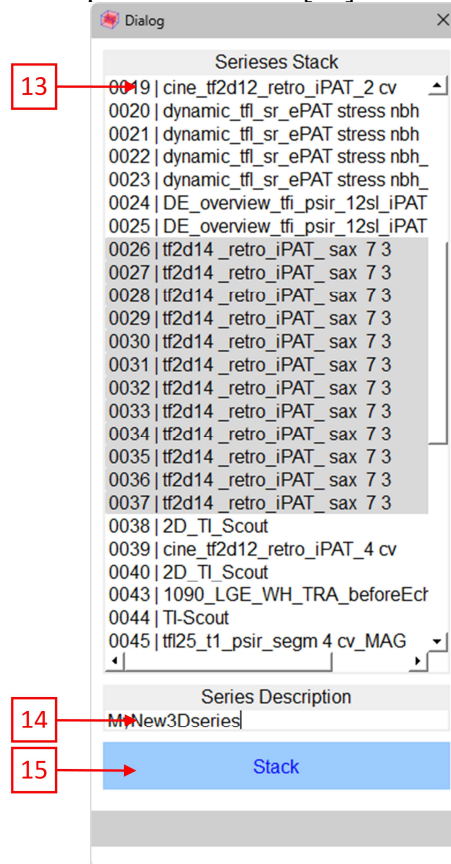

Figure 2: Reslice3Dto2D stack data to a 3D set

The stacking requires to choose the data that should be stacked [13] and a series description [14]. After choosing accordingly, the data can be stacked [15] if it passes the internal check. The chosen data must have the same number of phases, parallel slices and the same distance if more than two serieses are selected. The stacked data can be exported from the main GUI [12]. It is important to stack before performing the reformatting as otherwise the series numbers double. The stacked data can be used for reformatting, however in case of parallel imaging or subsequent measurements of a short axis stack, the stacked data may not align in the phase and/or have a significant higher resolution in the third dimension than the in-plane pixel spacing.