Instructions for Use

Trueview 2.12.0 Software Development Kit

Table of Contents

1. Description of the device 3
   1.1. Software 3
   1.2. Common assets 3
   1.3. API 4
   1.4. Code examples 4
   1.5. License Manager 4
   1.6. Demonstrator 4
2. Intended Use and Indications for Use 4
   2.1. Intended Use 4
   2.2. Intended User 5
   2.3. Indication for Use 5
   2.4. Restrictions on Use 5
   2.5. Intended Use Environment 6
3. Cleaning and sterilisation 6
4. Performance and accuracy 6
5. How to use the device 7
   5.1. Minimum hardware and system requirements 7
   5.2. Security configuration 7
   5.3. Setting up the Trueview SDK for development 7
   5.4. Integrating with the Trueview API 8
      5.4.1. Generic instructions for all specialisms 8
      5.4.2. Scatter correction and bone health specialisms 9
      5.4.3. CBCT and DRT reconstruction specialisms 9
      5.4.4. Code examples 10
      5.4.5. Technical documentation 10
   5.5. Installing Trueview on a production machine 10
   5.6. Licensing 11
   5.7. Update or uninstall the device 12
6. Risks and cautionary notices 13
   6.1. Safety risks 13
   6.2. Security risks 13
7. Packing and transport 14
8. Symbols, Labelling and Manufacture 15
   8.1. Label 15
   8.2. UDI 15

8.3. Manufacturer 16
9. Emergency situations 17
10. Document approvals 17
11. Revision history 18

1. Description of the device

For the purpose of these instructions, the device consists of the Trueview software and its Software Development Kit (SDK). The Trueview SDK contains the following items:

- The Trueview software runtime libraries (DLLs) and their dependencies.
- The Trueview common assets required for running the Trueview software.
- The Trueview Application Programming Interface (API) headers for the C and C++ languages and a CLR assembly to be used with languages supporting the .NET framework.
- The Trueview code examples demonstrating the usage of the API in multiple languages (C++ and C#) for supported Trueview specialisms.
- The Trueview License Manager tools which enable the management of the Trueview license.
- The Trueview Demonstrator application.

In what follows, we refer to the "development machine" as the computer on which the application using the Trueview software is built, and the "production machine" as the computer that will be running Trueview in production. Instructions on how to setup Trueview for both development and production environments can be found in sections 5.3 and 5.5.

1.1. Software

The Trueview software offers functionalities to improve, enhance, and exploit data from X-ray imaging. These functionalities are offered in multiple "specialisms":

- **Digital Radiography** (DR),
- **Mammography** (Mammo),
- **Bone Health** (BH),
- **Cone Beam Computed Tomography** (CBCT),
- **CBCT Reconstruction** (CBCT-Recon),
- **DR Tomosynthesis Reconstruction** (DRT-Recon).

Please consult section 2 for more information on the features offered in each specialism. Trueview functionalities for all specialisms are provided by the file `Lib\IbexSDK.dll` and its dependencies, and `Lib\IbexSDK.NET.dll` for the CLR interface. The dependencies are all the other DLLs listed in the same directory excluding some dependencies documented in section 5.3. All these DLLs must be present on the machine for Trueview to operate.

1.2. Common assets

The Trueview common assets are binary data required for the general operation of the Trueview software. This currently consists of the files in the `Lib\Segmentation` folder. For the DR, BH, CBCT-Recon, and DRT-Recon specialisms, as well as for the CBCT specialism without forward projection support, this folder must be present on the machine for Trueview to operate, and its folder structure

relative to the `IbexSDK.dll` file must not be altered (both in development and production). Assets may be added or removed from this folder in future releases of the software.

1.3. API

The Trueview API provides programmatic access to the Trueview software functionalities. The `IbexSDK.NET.dll` file contains all the CLR assembly information required for programming against this API with languages supporting the .NET framework. Other languages such as C or C++ should link directly to `IbexSDK.dll` and require additional textual headers to access this API, which are listed in the `Include` folder. For the C language the header is named `IbexTrueview.h` and for the C++ language the header is named `IbexTrueviewCPP.h`. These header files also contain documentation on all functions required for accessing the Trueview software functionalities, although this documentation is most easily accessed in the HTML format, also provided in the SDK (see section 5.4). Headers are only required on the development machine.

1.4. Code examples

The Trueview SDK code examples demonstrate how to integrate the various parts of the Trueview API together to form an application. These examples are provided in the `Examples` folder. More information on integration can be found in section 5.4.

1.5. License Manager

The License Manager is a standalone utility that can be used to license the Trueview software for development or production machines, including machines that do not have an internet connection. It comes in two forms: a command-line utility, and a graphical utility. See the section 5.6 for more information on licensing.

1.6. Demonstrator

The Trueview demonstrator only showcases the DR, Mammo, CBCT, and BH Trueview specialisms. This graphical application performs end-to-end scatter correction of a radiograph, mammograph, or CT scan and, for the BH specialism, the measurement of the bone health in selected regions. It should only be used for demonstration purposes. This application is not covered further by this document.

2. Intended Use and Indications for Use

2.1. Intended Use

The device has several intended uses, depending on which specialism is used:

- **Scatter correction.** This is provided in the DR, Mammo, BH, and CBCT specialisms. Intended use: produce scatter-corrected radiographs or mammographs (herein referred to as images) without using an anti-scatter grid (ASG).
- **Bone health.** This is provided in the BH specialism only. Intended use: the Trueview software Bone Health functionality (BH) is intended to provide opportunistic screening for osteoporosis from digital radiographs. The BH functionality analyses image data from a digital radiograph to generate outputs pertaining to the diagnosis and treatment of osteoporosis, including a measurement of areal bone mineral density (aBMD) at the examination site. These measures provide indications which can be used by qualified interpreting clinical professionals to support case finding and diagnosis of osteoporosis.

- **3D reconstruction.** This is provided in the CBCT-Recon and DRT-Recon specialisms only. Intended use: Produce a reconstructed 3D data set (or volume) from a set of X-ray images created during a CBCT or tomosynthesis scan.

2.2. Intended User

The immediate intended user of this medical device is a software developer (or "integrator"), who wants to integrate the Trueview software with their medical software through the Trueview API. For a safe and secure integration, the software developer requires sufficient skills in the programming language chosen to interface with the Trueview API (C, C++, or C#) and should be suitably trained in state-of-the-art cybersecurity best practices. The deployment of the resulting software further requires skills in software deployment, administration, and maintenance, as appropriate for the chosen deployment environment. No other domain-specific skills are required.

The intended end-user is a radiographer or a radiologist, who is using the medical software created by the integrator. It is therefore the responsibility of the integrator to handle interactions with the end-user. For Bone health, the assessment of the output must be made by a qualified health care practitioner.

2.3. Indication for Use

The use of the Trueview software functionality is indicated in the following situations:

- **Scatter correction.** Used as post-processing step of a traditional DR, CT, or mammography workflow, in cases where the use of an ASG is impossible or impractical, or when the removal of the ASG from an existing system would provide efficiency improvements (e.g., reduced dose, reduced maintenance, or easier operation).
- **Bone health.** Used as post-processing step on wrist scans in PA orientation (palm down on the detector, single wrist, see clinical guide for use), opportunistically for patients at risk of reduced bone density, where an early indicator of bone health would enable more efficient clinical decision making (e.g., referral to DEXA).
- **3D reconstruction.** Used as post-processing step of traditional CT scans, full or half-fan, with arcs of 180 to 360 degree, or in DR tomosynthesis scans, where the data taken during the scan must be used to produce a 3D data set for further inspection.

2.4. Restrictions on Use

The use of the Trueview software functionality is explicitly *not* indicated in the following situations:

- **Scatter correction.** The functionality must not be used on images obtained with an ASG. In addition, in the mammography specialism, the functionality must not be used on mammographs other than anatomical, biopsy-mode, spot-compression-mode, and specimen-mode.
- **Bone health.** The functionality must not be used on images obtained with an ASG, on images of body parts other than a single wrist, or on wrist images with orientation other than PA.
- **3D reconstruction.** The functionality must not be used on CT scans other than circular scans where the detector is perpendicular to the source-to-centre-of-rotation vector, and with either the detector rows or columns aligned with the plane of rotation. In particular, using this feature for helical CT scans is not indicated. The functionality must not be used on tomosynthesis scans other than scans where the source and detector are moving on a

straight line, and with the detector rows or columns aligned with the direction of motion of the source.

All the functionalities require calibration of the X-ray system, and they must not be used on a system (or system setup) which falls outside of the calibrated regime. The calibration procedure is not handled by the Trueview software, hence is not covered in depth in this document. See section 4 for more information.

2.5. Intended Use Environment

This software device is not standalone and must be compiled into another software device (the "final standalone product"). The intended use environment of the final standalone product is a controlled clinical environment (e.g., a clinical practice or a hospital). The final standalone product is intended to run on a workstation operated by a healthcare professional (e.g., radiologist, field engineer, or healthcare provider IT staff), or on a server.

3. Cleaning and sterilisation

There are no sterility requirements as this is a software-only device.

4. Performance and accuracy

In the BH specialism, the software runs an outlier detection algorithm that outputs an "outlier" flag. This flag is raised whenever a given data set (inputs and outputs) deviates significantly from expectations. In such case and for elective scans, the operator should be advised to double-check their inputs, as well as review intermediate outputs of the software (such as the automatically detected regions of interest).

Otherwise, no measurement of the accuracy of the outputs are made by the software. The end-user operator is responsible for verifying the quality of the outputs, and for determining if an output is of diagnostic quality.

Calibration must be performed prior to using the Trueview software.

- For the scatter correction, the calibration process consists of taking a series of radiographs of manufactured samples at varying thickness levels using the settings for each of the imaging protocols. These calibration samples are not part of the product. Once the calibration radiographs are obtained, they must be sent to IBEX to create a calibration database for the chosen imaging protocol. This database enables the Trueview scatter-correction and bone health functionalities for radiographs obtained with this protocol.
- For the BH specialism, the outlier detection feature requires additional calibration; if not calibrated, the outlier flag will not be reliable and should be ignored. The bone mineral density predicted by the software also requires mapping to an existing DXA system. A default mapping will be supplied, however the mapping data can be updated to meet customer-specific requirements if necessary.

Please contact IBEX for more information on the calibration process.

Periodic recalibration of the system may be required to compensate for changes in the X-ray system. The corrective action frequency is dependent on the stability of the X-ray system, so service intervals cannot be specified here.

5. How to use the device

5.1. Minimum hardware and system requirements

The Trueview SDK is designed to run on hardware meeting or exceeding the following specifications:

- Operating system: Windows 10 64bit Professional version 1607.
- CPU: Intel i7 multi-core, or equivalent from other manufacturers.
- RAM: 32GB.
- Storage: 5GB free space (before installation).
- GPU: Nvidia card with 8GB of on-board memory, CUDA compute capability 6.1 (e.g., GeForce GTX 1070, or Quadro P5000).
- GPU drivers: Nvidia driver 522.06 (October 2022).

Operating the Trueview SDK on hardware that does not meet these specifications is not supported.

Any other component required to run the Trueview SDK on the system will be automatically installed by the Trueview installer. If the installer is not used, refer to section 5.3 for detailed instructions.

5.2. Security configuration

The Trueview SDK has no configurable security options. See section 6 for security recommendations for the integration into the operating environment.

5.3. Setting up the Trueview SDK for development

To be able to develop an application using the Trueview SDK, the Trueview installer must be configured to install at least the following features:

- Trueview ® SDK (*mandatory*).
  - .NET Support (*recommended*).
    - License Manager (*recommended*).
- If integrating with the C or C++ interface:
  - C/C++ Development (*mandatory*).
    - Examples (*recommended*).
    - Documentation (*recommended*).
- If integrating with the CLR interface:
  - .NET Support (*mandatory*).
    - Examples (*recommended*).
    - Documentation (*recommended*).

This will install the files (DLLs, headers, import libraries, …) required for compiling programs making use of the Trueview API. The examples and documentation, although optional, are a recommended resource for learning how to use the Trueview API; see section 5.4. The “License Manager” feature will install the License Manager tools, which can assist in licensing Trueview on the development machine; see section 5.6.

The installer adds the `Lib` folder to the `PATH` environment variable. If this folder needs to be moved for any reason after Trueview is installed, this environment variable must be updated. If the `PATH` environment variable cannot be updated or used, it is the responsibility of the integrator to ensure that all the DLLs required by Trueview can be found in the DLL search path of their application, at all

times. For example, this can be achieved by placing these DLLs in the same directory as the application executable.

The Trueview SDK will assume that the common assets and the license file are located in the same folder as the `IbexSDK.dll` file.

Because of limitations on path lengths in Windows, the Trueview SDK cannot be installed or moved to a location that would lead to path lengths longer than `MAX_PATH` (260 characters). The Trueview installer will not allow installing the SDK at such locations, and the integrator must observe the same restriction if they manually move the SDK to an alternative location and rely on the `PATH` environment variable.

To prevent Windows from timing out long-lasting GPU computations, hence interrupting the Trueview processing, the Trueview installer modifies (or creates) the registry key `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\GraphicsDrivers\TdrDelay` and sets it to a large value (360 seconds).

Warning: This registry key may be overwritten when updating graphics card drivers, and care must be taken to ensure it always remains set to a large value.

The Trueview SDK is built against CUDA 11.8 (Update 1). This requires an Nvidia graphics card with up-to-date GPU drivers (see section 5.1). Both "Standard" and "DCH" drivers are supported. However, to support "Standard" drivers, the Trueview installer will add the `%ProgramW6432%\NVIDIA Corporation\NVSMI` folder to the `PATH` environment variable to enable loading the file `nvml.dll`.

The Trueview SDK and its dependencies require the Microsoft Visual Studio C++ 2022 redistributable. This will automatically be configured by the installer. If copying the Trueview SDK DLLs to another machine, care should be taken to ensure that this new machine also has the appropriate redistributable installed. To use the CLR interface, the .NET Framework v4.6.2 should also be installed (this is installed by default starting from Windows 10 version 1607).

At runtime, the Trueview SDK will attempt to write cache files to the `%LOCALAPPDATA%\Local\Temp\ArrayFire` folder. If this directory is accessible and writable, the cached data will enable Trueview to run at optimal performance.

Warning. In a fresh setup, the above cache will not yet exist, and the first execution of Trueview will need to create it. This will incur a performance penalty. This will only affect noticeably the first run of Trueview, and the few follow up runs in a lesser extent, until the cache is fully built. The runtime performance of Trueview during the first runs will therefore not be representative of the performance in normal use.

5.4. Integrating with the Trueview API

5.4.1. Generic instructions for all specialisms

The general workflow of the Trueview API can be summarized as follows:

1) Initialise the Trueview Context. This must be performed before any interaction with the Trueview API.
2) Load and process inputs with the Trueview API.

3) Destroy the Trueview Context. This must be performed only when the Trueview API is no longer needed, e.g., on application shutdown.

It is optimal to initialise the Trueview Context only once in an application rather than to repeatedly initialise and destroy it for each processing job.

The Trueview API does not support any form of concurrent access. All interactions with the Trueview API should be carried on in the same thread that created the Trueview Context, and only a single process should be using the Trueview API at any point in time on a given machine.

5.4.2. Scatter correction and bone health specialisms

Assuming a Trueview Context is already created (see section 5.4.1), the workflow of the scatter correction and bone health API can be summarized as follows:

1) Load inputs for the chosen protocol. This must be performed each time a new protocol is selected (or once when the application starts, if there is only a single protocol available).
2) Run the scatter correction process for one or more images obtained with that protocol. This and following steps can be repeated for as many images as required, provided they were obtained with the same protocol. If a different protocol must be used, go back to step 1.
3) BH only: Run the automated region detection on the image. This outputs all known region-of-interests visible on the image.
4) BH only: Using the outputs of the scatter correction and region detection processes, run the bone health measurement process. This combines the outlier detection process and the measurement of bone health indicators, which are automatically mapped to an areal bone mineral density and a T-score.

The inputs required for a given protocol are provided by IBEX and include calibration databases (specific to the X-ray device being used, see section 4), scatter correction databases, and processing settings. For the BH specialism, this also includes a validation database for outlier detection. The databases can be loaded and selected automatically based on the capture conditions of the input images. Except for the BH specialism, the databases can also be loaded and selected manually in the API.

The input images must be dark-field-corrected and flat-field-corrected, and they must not contain any invalid pixel (infinite, not-a-number, negative, or zero). The input images can optionally be dose-normalised (i.e., the image is divided by the dose, or mAs). If the input images are not dose-normalised, the dose must be provided to the Trueview API. Details on how to load these data in a form suitable for the Trueview API and how to launch the scatter correction procedure are provided in the examples.

5.4.3. CBCT and DRT reconstruction specialisms

Assuming a Trueview Context is already created (see section 5.4.1), the workflow of the CBCT and DRT reconstruction API can be summarized as follows:

If using scatter correction:

1) Load inputs for the chosen protocol. This must be performed each time a new protocol is selected (or once when the application starts, if there is only a single protocol available).

2) Process one or more scans with that protocol. This can be repeated for as many scans as required, provided they were obtained with the same protocol. If a different protocol must be used, go back to step 1.

If not using scatter correction:

1) Process one or more scans. This can be repeated for as many scans as required.

The output volumes will be provided in Hounsfield units for CBCT scans, and absorption coefficients for DRT scans, unless otherwise specified by settings.

5.4.4. Code examples

The C# examples can be accessed by opening the Visual Studio solution file `Examples\CSharp\Examples.sln`. For detailed information, please refer to:

- **(CBCT ONLY)** The ConeBeamComputedTomography project.
- Other modalities: C# examples are not provided at the moment.

The C++ examples are listed in the `Examples\CPP` folder. They are configured using CMake, which can be used to produce a Visual Studio solution file or to produce build scripts for any other build system supported by CMake. The CMake scripts use the `TRUEVIEW_ROOT_DIR` variable to locate the Trueview API (i.e., the folder containing the `Include` and `Lib` sub-folders). By default, this variable is initialised assuming the standard folder structure created by the Trueview installer. Therefore, no extra command line argument needs to be provided to CMake unless the example files have been copied to an alternative location. The generated executables assume that the `Lib` folder has been added to the `PATH` environment variable. For detailed information, please refer to:

- **(DR ONLY)** The file `ScatterCorrection\DigitalRadiography\DigitalRadiography.cpp`.
- **(BH ONLY)** The file `ScatterCorrection\BoneHealth\BoneHealth.cpp`.
- **(MAMMO ONLY)** The file `ScatterCorrection\Mammography\Mammography.cpp`.
- **(CBCT ONLY)** For integrating Trueview with forward projection data, see the file `ScatterCorrection\ConeBeamComputedTomography\ForwardProjectionRun.cpp`. Otherwise, see the file `ScatterCorrection\ConeBeamComputedTomography\StandaloneRun.cpp`.
- **(CBCT-RECON ONLY)** The file `Reconstruction\ReconConeBeamComputedTomography\ReconConeBeamComputedTomography.cpp`.
- **(DRT-RECON ONLY)** The file `Reconstruction\ReconDigitalRadiographyTomosynthesis\ReconDigitalRadiographyTomosynthesis.cpp`.

5.4.5. Technical documentation

In addition to code examples, the Trueview SDK also provides detailed technical documentation for all the public API functions. This documentation can be found in the `Documentation` folder.

5.5. Installing Trueview on a production machine

To be able to run an application developed with the Trueview SDK, the Trueview installer must be configured to install at least the following features:

- Trueview ® SDK (*mandatory*).
  - .NET Support (*recommended*).

        - License Manager (*recommended*, or *mandatory* when no internet connection is available).
- If Trueview is integrated into a CLR application:
    - .NET Support (*mandatory*).

This will install the DLLs required for running an application making use of the Trueview API. The "License Manager" feature will install the Trueview License Manager, which is required if the machine has no internet connection and has not already been licensed to use the Trueview SDK. This tool may also be necessary if the software that integrates the Trueview SDK does not provide its own solution for licensing. See the section 5.6 for more information.

The instructions from section 5.3 regarding the `PATH` environment variable, the location of the Trueview SDK DLLs, the restrictions on path lengths, the required modifications to the Windows registry, the GPU drivers, the Microsoft Visual Studio redistributable, and the .NET Framework also apply to production machines.

Warning. As discussed in section 5.3, the very first execution of Trueview will need to create cache files in the `%LOCALAPPDATA%\Local\Temp\ArrayFire` folder. This will incur a one-time performance penalty, and can be an issue if consistent, optimal performance is a critical requirement. In such cases, pre-built cached data can be copied onto the production machine from another machine with the same GPU device and drivers, using the above location.

5.6. Licensing

The Trueview software requires a machine to be licensed before that machine can use any of the software's functionalities. IBEX provides three types of licenses: demonstration, development, and production licenses. Demonstration and development licenses are typically capped to a small number of activated machines and have a fixed expiration date. Conversely, production licenses are not capped and will never expire. The production license details will normally be sent at the end of the integration process, once both parties are satisfied with the integration and the product has been validated for use in production.

Warning. As their name implies, demonstration and development licenses are not suitable for use on production machines. These licenses will expire, which will cause the software to refuse to run while in production. Furthermore, these licenses are not invoiced at the same rate as production licenses, and their use in production can constitute a violation of contract. Production machines must therefore use a production license.

The Trueview licensing involves creating a license file for the machine running the software. This file is checked by the Trueview software to verify the license validity before the software's functionalities can be accessed. The procedure to generate this file depends on whether the machine has access to the internet or not. If internet access is available, the Trueview software can use a fully automated procedure which is documented in the code examples. This procedure requires providing a license key and token to the Trueview API. Otherwise, the License Manager utility must be used. More information on the licensing procedure can be found by running this utility (with the `-Help` command line option if using the command-line version), and by reading the licensing quick start guide provided alongside this document.

Once a license file is created, no internet connection is required until the license needs re-validating. A license needs re-validating if it expires (only applicable to demonstration and development licenses), or if the machine's hardware changes significantly since the file was first created. If no internet access is available when the license needs re-validating, the License Manager utility must be used.

The license may restrict usage of the software inside a virtual machine of a sandboxed container. If such is the case, an additional License Agent tool is provided with the installer. This tool must be launched outside of the virtual machine of sandboxed container (on "bare metal"), and it must be configured with a valid license. The tool is then able to serve the license to the software which is using the Trueview SDK and which is running inside a virtual machine or container. To achieve this, the Trueview API must be used to explicitly request a connection to the License Agent; see the technical documentation. This connection is established using the HTTP protocol over standard TCP/IP, and must remain alive for the duration of the Trueview processing.

In addition to the Trueview license, users of the Trueview SDK are bound to the licenses of the third-party dependencies used by the Trueview SDK. These can be accessed in the `Licenses` folder created by the installer, along with the Trueview End User License Agreement (EULA). Some of these third-party licenses have specific requirements that must be fulfilled by users of the Trueview SDK. The most common requirement is to include the text of the corresponding third-party license in any distribution of derivative products (which includes any product built against the Trueview SDK). Other requirements may apply; please refer to the individual licenses for more information.

5.7. Update or uninstall the device

The installer can be used to perform updates or to uninstall the Trueview software or its SDK. Updates require running a new installer which will be provided by IBEX. The installer software implements security controls to ensure the integrity of the update, and it is therefore the recommended method for deploying updates to the Trueview software. If the installer is not used, it is the sole responsibility of the integrator to ensure that the updated files are deployed successfully and without data corruption.

As discussed in section 5.3, the Trueview SDK writes cache files to the `%LOCALAPPDATA%\Local\Temp\ArrayFire` folder. It is recommended to clear this cache in the following situations: when graphics card drivers are updated, or when the graphics card is physically replaced by another model.

Warning. If the cache is cleared after an update, the runtime performance of Trueview will be noticeably impacted until the cache is fully re-built (see sections 5.3 and 5.5). Therefore, this operation should only be performed in an environment where runtime performance is not critical. Alternatively, an updated cache can be installed following the procedure discussed in 5.5.

Warning. When updating or uninstalling the Trueview software, make sure no application is running on the operating system that could make use of the Trueview software. This includes applications built with the Trueview API, as well as development environments that are used to build such applications (e.g., Visual Studio).

Warning. When updating or uninstalling the Trueview software, any change made to the code examples will be lost. It is recommended to not modify the code examples installed with the software, and instead create and work on a copy of this code to avoid this issue.

6. Risks and cautionary notices

6.1. Safety risks

1. The software device should run on a computer (or virtual machine) that is independent and not responsible for operating the X-ray equipment. If these instructions are followed, this device may be used safely without risk to patient or radiographer.
2. If the Trueview software fails to process a given input for any reason, no output will be generated. The Trueview API or the operating system will report such failures, and it is the responsibility of the integrator to capture and act on these errors. In particular, it is the responsibility of the integrator to back up the original input data in case an unrecoverable error forces the process to close.
3. The Trueview software processing can be influenced by a series of settings which are accessible through the API. These settings have default values that have been tested and are appropriate for safe processing. If the device integrating the Trueview software allows the end user to change the settings, appropriate controls should be introduced to ensure that the settings are not modified in a way that would either compromise the quality of the output or introduce unreasonable resource usage on the workstation. For example, once the integrated device is deployed, access to the settings should be limited to an appropriately safe subset, and access could also be further limited to authorised users only.

6.2. Security risks

High level summary: The Trueview software does not handle any sensitive information. The services rendered by the Trueview software are not safety critical. If an attacker only disrupts operation of the software without impacting the broader operating environment, no harm can come to the patient or operator. The security risks and recommendations below are primarily meant to limit risk to the operating environment.

1. The device is able to use an existing connection to the internet for the purpose of license validation, however this is not a requirement for the operation of the device. The associated cybersecurity risks can be mitigated by disabling internet access and using the offline license validation methods described in section 5.6.
2. The device offers interfaces for loading files from the file system (e.g., TIFF images and proprietary binary files supplied by IBEX). While these interfaces are monitored by IBEX for known vulnerabilities, they may still be the source of an attack. Therefore, these interfaces should not be used to load untrusted files. This includes files sent to the device from untrusted sources or loaded from an untrusted location on the file system or the network.
3. Once deployed, the device does not require special privilege for its operation, aside from the list below. No other privilege should be granted to the device, to mitigate the risk of privilege escalation in case of an attack.
    - read access to its installed location and the operating system libraries,
    - write access to the configured directory, for the storage of logs, and
    - write access to the directory hosting the main SDK library, for the creation of the license file.

4. Once deployed, the shared libraries used by the device should be protected against accidental or malicious modification, for example by marking them as read-only or opened to modification only by a user with elevated privilege.
5. To reduce the risk of attacks, the device should be used in an operating system that is kept up to date with security patches from the manufacturer.
6. To reduce the risk of attacks, the device should be used in an operating system that is secured with state-of-the art protection, such as user access controls following the least privilege principle, anti-virus, and anti-malware protections.
7. To reduce the risk of attacks, the device may be executed in a sandboxed environment (e.g., virtual machine or container) such that it cannot affect its operating environment even if compromised.
8. The device is intended to be used by a qualified professional, in a controlled environment implementing the security features and procedures listed above. If operated outside of this scope, the device may be used as a point of entry in an attack, and could be used to tamper, disrupt, or abuse the operating environment.

7. Packing and transport

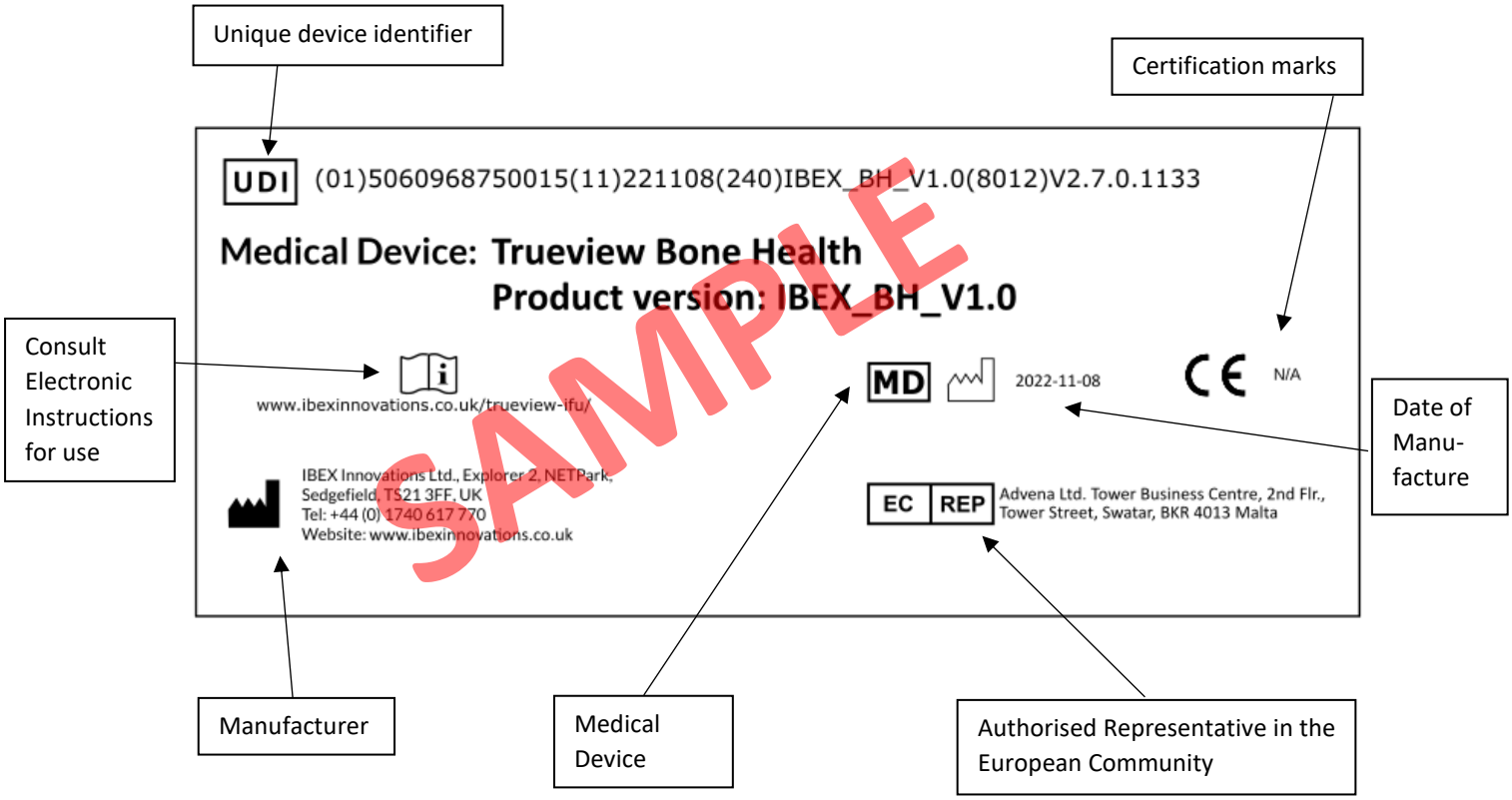
Not applicable.

8. Symbols, Labelling and Manufacture

8.1. Label

The labels to be used with this version are included in the `Lib` folder. An example is shown below, highlighted with the symbol meanings. These labels must be made available to the end user of the software, either in a printed form, or displayed in electronic format.

For the US: Trueview is intended to be used in a clinical setting, therefore, the following labelling will also be added: “Rx only – For prescription use only”.



8.2. UDI

Each specialism included in the Trueview Software is assigned its own Unique Device Identifier (UDI). The UDI can be found on the respective label (see above) and can also be queried from the Trueview API. See the technical documentation for more information. For example, the list of UDIs (including UDI, product name, and EC REP) for all the products enabled on the current license can be obtained with:

- C++ API: `Ibex::Base::Context::GetUDIs()`.
- C# API: `Ibex::Base::UDI::GetUDIs()`.

8.3. Manufacturer

	IBEX Innovations Ltd., NETPark Plexus, Thomas Wright Way Sedgefield, TS21 3FF, UK Tel: +44 (0) 191 249 7912 Website: https://ibexinnovations.co.uk/

9. Emergency situations

In the event of a serious incident (leading to death or serious deterioration of health) involving this device, the user must inform IBEX (the manufacturer), and the competent authority of the member state in which the user is established as soon as possible. When in doubt, report.

A list of contact points of the competent authorities for EU member states can be found here: https://health.ec.europa.eu/system/files/2022-10/tse_bse_contact_points.pdf.

10. Document approvals

This document approved by:

Software Lead:	Corentin Schreiber	**Signature:**		**Date:**	Mar 3, 2025
Quality Manager:	Kurt Scott	**Signature:**		**Date:**	Mar 3, 2025

11. Revision history

Version number	Changes made	Author(s) initials
1.0	Original release for Trueview version 1.2.0	KH
1.1	Updated for Trueview version 1.4.0	KH, CS
1.2	Addition of user description	KH
1.3	Updated for Trueview version 1.4.1	CS
1.4	Updated for Trueview version 1.4.2	CS
1.5	Updated for Trueview version 1.5.0. Added limitation of file path lengths. Added information on alternative loading of protocol data. Added limitation on multi-threading.	CS
1.6	Updated for Trueview version 1.5.1. Added more information on environment variables when using the CLR interface. Added information on TdrDelay registry key.	CS
1.7	Updated for Trueview version 1.6.0. Updated for removal of CommonLibs folder, installer feature for system environment, IBEX_TRUEVIEW_PATH environment variable, and Windows GAC. Added instructions for third-party software licenses.	CS
1.8	Updated for Trueview version 1.6.1	CS
1.9	Updated for Trueview version 1.6.2. Added information on Visual Studio redistributable. Fixed formatting.	CS
1.10	Updated for Trueview version 2.0.0. Updated dependency information to show that VS C++ 2015 runtime is no longer needed. Clarified what happens when a Trueview license needs re-validating. Added more precise pointers to the relevant example code for each modality. Updated after splitting the C++/C interface and .NET Framework interfaces into separate DLLs. Added more information on the impact of the ArrayFire cache, and how to mitigate the impact. Updated for inclusion of CBCT into the demonstrator. Updated after renaming BMD to Bone Health. Updated the label.	BR, CS
1.11	Updated for Trueview version 2.1.0. Added information on the License Manager GUI. Updated after changing the structure of the installer. Updated to mention the new technical documentation. Added Mammo-specific intended uses.	CS, BR

1.12	Added minimum hardware requirements. Updated risks and hazards section for clarity. Minor improvements to the wording.	CS, CM
1.13	Updated for Trueview version 2.2.0. Added instructions for use for the DRR specialism. Added table of content.	CS
1.14	Updated for Trueview version 2.2.1.	CS
1.15	Updated for Trueview version 2.3.0. Added cautionary notice about inputs and outputs not being backed up. Added information about the new CBCT-Recon specialism. Updated the location of the code examples.	CS
1.16	Updated for Trueview version 2.4.0. Spell checking. Clarified intended use, added indication for use. Clarified who is the end user. Moved section 6 to section 2.4.	CS
1.17	Updated for Trueview version 2.5.0. Added cybersecurity risk mitigation advice in section 6. Updated intended use and indication for use for 3D reconstruction to include tomosynthesis. Added manufacturer contact details in section 10. Added warning about use of non-production licenses for production machines in section 5.4. Added a sentence in section 5.5 to explain when production licenses are issued.	CS, BR
1.18	Updated for Trueview version 2.6.0. Added missing statement that the Segmentation folder is required for CBCT-Recon and DRT-Recon. Fixed spelling error. Updated Bone Health workflow, and added description of the outlier flag. Added warning about operating on machine with more than one CUDA-enable device.	CS
1.19	Updated for Trueview version 2.7.0. Updated instructions for DRR, including the new auto-alignment feature. Updated sample label to the new format.	CS
1.20	Updated for Trueview version 2.8.0. Removed outdated warning for machines with two GPUs. Updated minimum GPU driver version to support CUDA 11.8.	CS
1.21	Updated for Trueview version 2.9.0. Updated the wording of the Bone Health specialism for clarity. Added paragraph in section 5.5 to describe the license agent.	CS, KS

	Changes ported from MDR audit for version 2.6.0: Updated intended use for BH CE-marking. Added section to describe how UDIs can be obtained from the API. Added “SAMPLE” watermark on the example label to avoid confusion. Added section on intended use environment. Renamed section “Accuracy” to “Performance and Accuracy”. Merged Sections “Risks”, “Hazard summary”, and “Cautionary notices” into a single section titled “Risks and cautionary notices” to reduce duplication of information. Removed output quality indicator and flag from the “Risk” section since this information is not a risk and is covered in the “Accuracy” section. Separated the “Risk” section into safety and security risks. Added mention of safety risk if processing settings are not controlled. Added mention of security risks other than access to the internet. Updated to add in section on what to do in the event of a serious incident occurring.	
1.22	Updated for Trueview version 2.10.0.	CS
1.23	Updated for Trueview version 2.10.1.	CS
1.24	Updated for Trueview version 2.10.2. Fixed incorrect CUDA version number.	CS
1.25	Updated for Trueview version 2.11.0. Removed DRR product, now discontinued. Added warning about losing changes to code examples on update.	CS
1.26	Updated for Trueview version 2.12.0.	CS

Instructions for Use - Trueview 2.12.0

Final Audit Report 2025-03-03

Created:	2025-03-03
By:	Corentin Schreiber (c.schreiber@ibexinnovations.co.uk)
Status:	Signed
Transaction ID:	CBJCHBCAABAA_V6Uq_9tjJDhhIueTNSUCXVF8cKDIPBh

"Instructions for Use - Trueview 2.12.0" History

Document created by Corentin Schreiber (c.schreiber@ibexinnovations.co.uk)
2025-03-03 - 11:52:54 AM GMT

Document e-signed by Corentin Schreiber (c.schreiber@ibexinnovations.co.uk)
Signature Date: 2025-03-03 - 11:53:57 AM GMT - Time Source: server

Document emailed to Kurt Scott (k.scott@ibexinnovations.co.uk) for signature
2025-03-03 - 11:53:58 AM GMT

Email viewed by Kurt Scott (k.scott@ibexinnovations.co.uk)
2025-03-03 - 1:14:27 PM GMT

Document e-signed by Kurt Scott (k.scott@ibexinnovations.co.uk)
Signature Date: 2025-03-03 - 1:14:38 PM GMT - Time Source: server

Agreement completed.
2025-03-03 - 1:14:38 PM GMT

Adobe Acrobat Sign