# **VideoKifu** v1.3.1

for Android™ devices

© 2017–2025 Andrea Carta & Mario Corsolini

URL: www.oipaz.net/VideoKifu.html

eMail: andrea.carta@mclink.it

mario@corsolini.net

# Indexes

## Index of contents

## Index of figures

**page**

# 1. Introduction

VideoKifu for Android[1] reconstructs the whole move sequence of a Go game from the video feed provided by the camera of a smartphone or tablet. It may save a time-lapse video and it creates SGF[2] files that may be published in real-time on the Internet, as in this example:

`https://videokifu.altervista.org/`

# 2. Requirements

The app needs Android 9.0 (Pie, API level 28) or higher, a camera with a resolution of at least 1280 × 720 pixels (although 1920 × 1080 pixels would be better) and a processor powerful enough to analyse at least two frames per second.[3]

A charger could be needed when the game's time controls are slow.

A sturdy tripod is also highly recommended.

# 3. How to install

Install in a smartphone or tablet the latest distribution of VideoKifu from Google Play.[4]

# 4. How to use

## 4.1 Quick-start guide

### Placing

The device has to be placed, on a tripod if possible, in "landscape" mode next to the goban (whose borders will appear parallel or so to the

[1] Android is a trademark of Google LLC.

[2] `https://www.red-bean.com/sgf/`

[3] At first. After the first white stone's detection the analysis will become about four times faster.

[4] `https://play.google.com/store/apps/details?id=com.videokifu.android`

frame's sides), as high as possible. The goban's grid must be completely visible (borders included) and cover most of the frame,[5] otherwise the app could not locate it.

**Resolution**

Camera's resolution may be set in the "Video source" section of App settings (§ 4.2.2 on page 8). It should be in between 720p and 1080p, never under 480p; that's because over 1080p frame analysis could take too much time, while under 480p moves' detection could be tricky. Should goban's lines be too thick a low resolution could work better. On the contrary, if the goban is placed far from the device and it takes up only a small part of the frame, a higher resolution is needed.[5]

**Focus**

The goban's grid must be well in focus throughout the whole game, and that should not change even if unrelated objects would come into view. That's the reason why focus should be locked on the goban, if possible, before the start of the recording.

**Light**

The goban must be uniformly lighted up; pay much attention to not changing too much the lighting throughout the game. Above all avoid leaving part of the goban in the shadow or, on the contrary, under bright lights that could cause glares and make the surface look almost white.

**Ruleset (§ 4.2.1 on page 8)**

Unless already done, it's better to set up the rules before the start of the game, especially the number of possible handicap stones.

**Grid confirmation**

Moves' recording begins when the goban's grid starting position is upheld. The grid will usually be drawn on the screen, above the real one, at first in red, then, if its position won't change, in different shades of green up to light green: touching the image at this time will let the recording begin.

[5] Frame clipping (§ 4.2.3 on page 12) may possibly be changed.

If the grid could not be detected, either at times or at all, moving the device, even by little, could help, as well as improving the lighting or the Frame clipping (§ 4.2.3 on page 12). Last chance: setting manually the grid position,[6] although this could prevent the app from keeping it in place during the game (were such the case it would be better to disable the automatic tracking[7]).

**Moves' recording**

It's better to let the recording begin just before the actual start of the game (but after the nigiri, if occurred, although the app should realize the situation and start the recording anew once emptied the goban). The app is capable of recording games already started, albeit with two restrictions: a possible manual setup of the handicap stones, if present, and no more than 18 stones already placed on the goban (in any case the moves' sequence will be almost random). Should there be more stones on the goban the recording won't likely begin soon, or even at all.

Pay attention to not moving either the goban or the device during the game: should that happen the app will immediately track (if automatic tracking is not disabled) small movements among frames. Even greater movements might be tracked, but a delay will be expected. If automatic tracking fails, it will be mandatory to manually adjust the grid position[6] as soon as possible, in order to prevent obvious mistakes in the subsequent recording.

When either the window Grid setting (§ 4.2.6 on page 14) or the window Moves' manual editing (§ 4.3.4 on page 17) is open, or the virtual keyboard (to insert alphanumeric data) is visible, or another app is active, the recording pauses, as shown in the Status bar (§ 4.3.1 on page 16). Remember that, especially if captures have been made, moves played during the pause might not be detected when the recording restarts and — if three or more have been played — could be detected in wrong

[6] Grid setting (§ 4.2.6 on page 14).

[7] By means of the specific option in "Sundries" section in App settings (§ 11 on page 11).

order. That's why such operations must be accomplished as soon as possibile.

**Stop recording** (§ 4.2.8 on page 14)

At the end of the game the recording should be stopped by means of the specific menu entry: thus further stones' movements (or removals), made in order to settle the score, won't be detected nor will affect the moves' list. If necessary, it will be possible to select the true last move of the game.

## 4.2 Settings menu's entries

By means of the settings menu (5 in figure 1 on page 9) several app's extra features may be accessed. Should a setup window open and require a confirmation, it will always be possible to close it and discharge any change either by means of the system <Back> key, or by selecting again the menu entry responsible for the window opening, or by opening another window.

### 4.2.1 Ruleset

It's possible to set up either some predefinite rules[8] or some customised rules. When choosing this last option a name could be assigned to the rules and the five fields defining them will be activated.[9]

It's also possible to set up the specific game handicap: "no handicap", "automatic" or 2 to 9 stones. The predefinite setup is "automatic", that should work in most cases; in any case setting up the correct games' handicap will improve first moves' detection.

### 4.2.2 App settings

The app settings belong to four groups.

**Broadcast online**

When this section is active it becomes possible to publish in real time

[8] "AGA", "Chinese", "Japanese", "GOE" or "NZ"

[9] "Handicap placement", "Ko rule", "Scoring method", "Suicide" and how a possible "Illegal move" has to be reported.

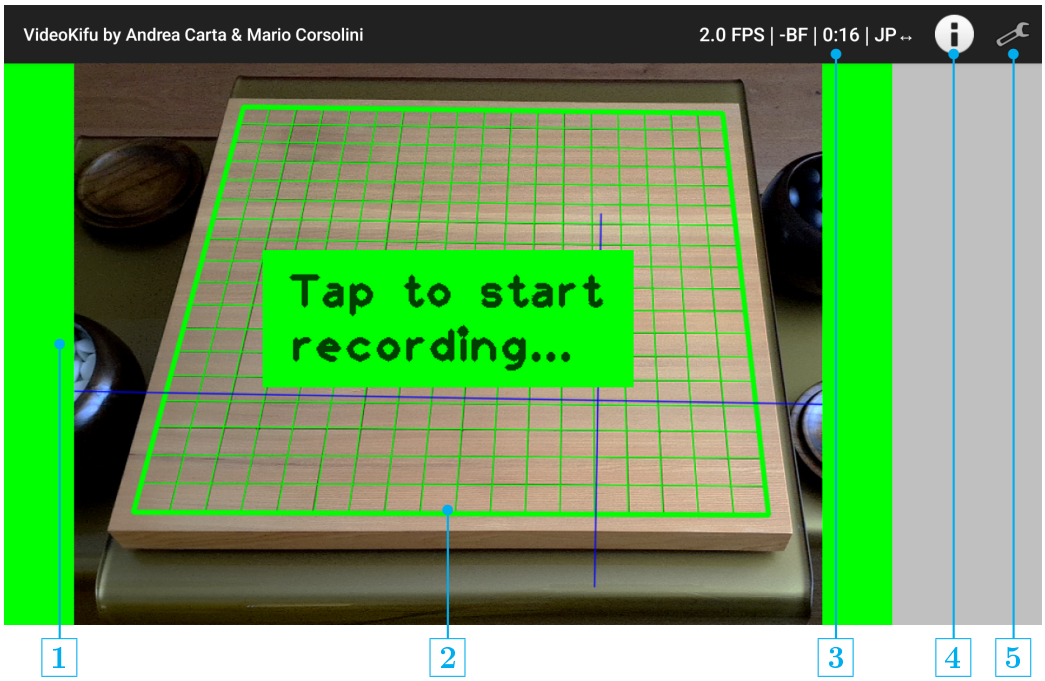


Figure 1: main window (before the start of the game).

1 Clipped frame's border.

2 Processed frame (green: detected grid; blue: ground line and vertical axis).

3 Status bar.

4 VideoKifu's infos (this guide included).

5 Settings menu.

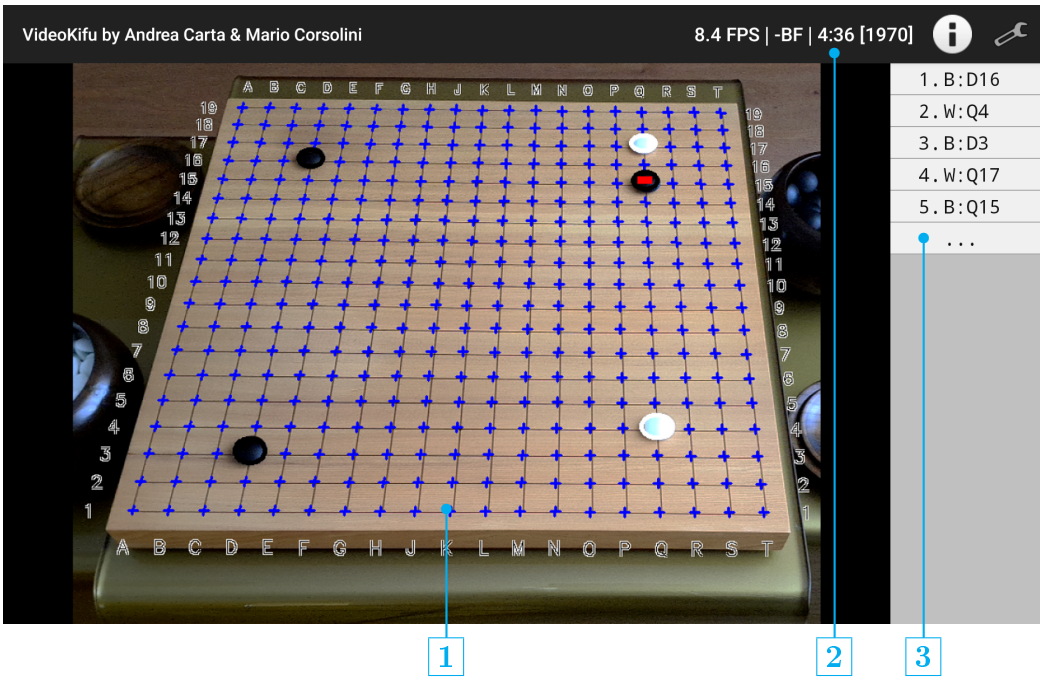


Figure 2: main window (after the start of the game).

1 Processed frame (highlighted: stones played, empty intersections and grid coordinates).

2 Status bar.

3 Moves' list.

on the Internet all the moves detected by the app. Preset settings of "Server" (ftp.videokifu.altervista.org), "User name" (vktv@videokifu), "Password" (VKtv4droid) and "Remote folder" (*empty*) allow the publication of the game[10] on `https://videokifu.altervista.org/`
Otherwise, the SGF file may be sent to any other website with FTP access. In such a case it would be better to use the button "Verify FTP settings" before the start of the game.
The alias of the publisher, as well as the point of view of related SGF files,[11] may be chosen as desired. If no selection is made for the latter, the app will use the point of view of Black, provided that it is possible to infer their position from the first moves.

**Frames filtering**

Should grid and/or stones' detection become difficult because of shadows/lights and so on, it's possible at any time to switch on/off some filters that will affect the image and could improve the app's performance: autogamma, sigmoidal contrast, stones' contrast and unsharp mask (whose further settings are: amount, radius and threshold, preset to 200%, 2.5 pixel e 5 levels).
First two options are preset at "on", last two at "off" (and any change to their status will be undone once closed the app).

**Video source**

It is possible to choose the camera and the native resolution of captured frames.

**Sundries**

It is possible to play a beep after each move is detected (its intensity could be adjusted by means of the device's volume controls).
Should grid's detection prove too problematic, automatic grid tracking can be deactivated.
If a recording of the game is needed, it's possible to save a series of time-lapse pictures and to merge them into a single video, at the end of the

[10] SGF and viewer/editor.
[11] The point of view set here will be the default one for local saves too.

game. Frames are saved after clipping and filtering, as the ones shown in the display of the device (without superimposed info). Since Android 10 (Quince Tart, API level 29), frames are saved into a subfolder named after the recording's start time (in UTC time zone) inside the folder "Pictures / VideoKifu", whereas the video is saved inside the folder "Movies / VideoKifu".

### 4.2.3 Frame clipping

The clipping border's colour (that is the rectangle between 1 e 2 in figure 3 on page 13), as well as the main window's border's colour in the early grid detecting phase, changes from red to yellow to green, depending on the clipping size. That's a warning because a small clipping could make difficult (or even impossible) detecting either the grid or the stones, or both.

### 4.2.4 Game details

This window's parameters are not mandatory when recording a game, but any info provided will be written in the SGF file, thus making easier to manage it afterwards. Should any info be provided when the game is still in progress, the recording will continue in background, unless a virtual keyboard should be needed on the screen.

After enough stones will have been detected,[12] the score of the game will become evaluable, and the specific button will become active. An estimation will be computed accordingly to the rules and Komi set up by the user, and will take a few to many seconds depending on the device; the recording won't be suspended.

### 4.2.5 Create SGF

At any time after the start of the recording it's possible to save on the device (besides Internet broadcasting, if the specific option[13] is set) the current SGF file, after choosing a suitable point of view. If several SGF files are created

[12] One third of the goban's intersections.

[13] "Broadcast online", in App settings (§ 4.2.2 on page 8).

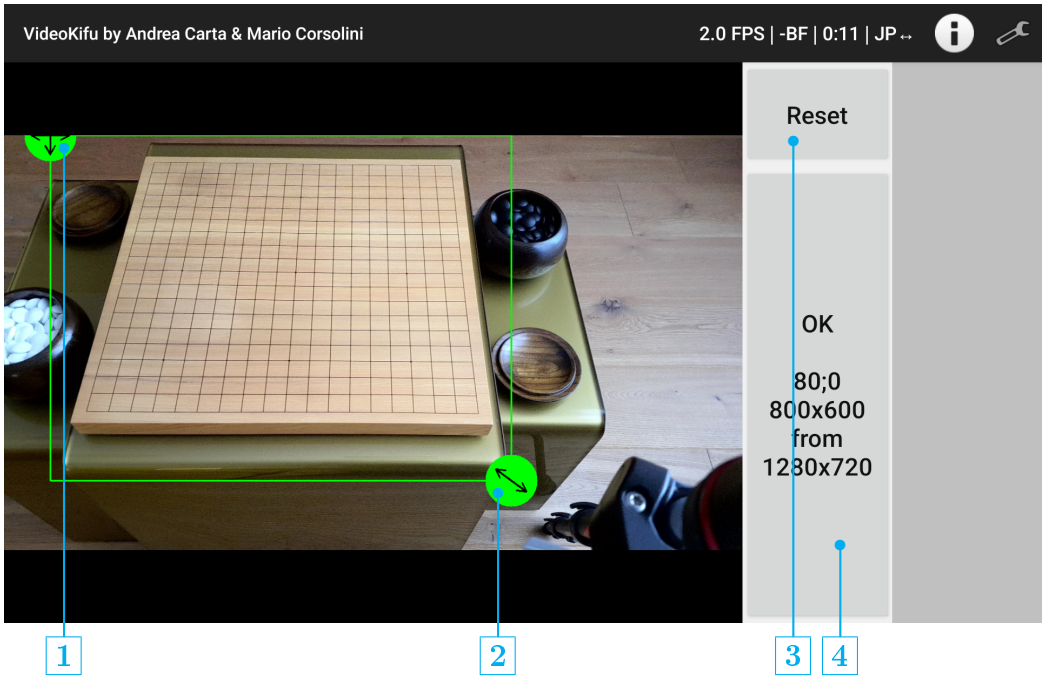


Figure 3: frame clipping.

1 Drag to move clipping area inside the frame.

2 Drag to expand/shrink clipping area (works only before the start of the game).

3 Sets the greatest centered clipping area (works only before the start of the game).

4 Upholds the clipping area set up at the moment. Informations shown are (down on): the clipping's upper left corner's coordinates; the clipping size; the original frames' size, as recorded by the device's camera.

from different points of view, all of them will be saved in the device's memory while only the last selected one will be published on line

Since Android 10 (Quince Tart, API level 29), such files are saved into subfolders named after the recording's start time (in UTC Time Zone), inside the folder "Documents / VideoKifu".

### 4.2.6 Grid setting

At first in the setting/editing grid window — the one visible in figure 4 on page 15 — no corner is selected, thus enabling a fast correction of a possible grid shift, should it be by an integer number of rows/columns.

To select (or deselect) a corner a long tap on the corresponding image will be needed.

A short tap will instead move the corresponding corner to the point touched.

### 4.2.7 Undo last edit

Restores the moves' list as it was before the last manual edit made by the user.

### 4.2.8 Stop recording

Stops the recording of the current game. Afterwards manual edits will still be possible (both to the moves' list[14] and to game infos[15]) as well as saving the last SGF file on the device or even on the Internet.[16]

### 4.2.9 Show report

It is visible only after the end of the recording. It shows again the final report on the files created by the app.

[14] As described in Processed frame (§ 4.3.2 on page 16) and also, only for certain edits, in Moves' list (§ 4.3.3 on page 17).

[15] Ruleset (§ 4.2.1 on page 8) and Game details (§ 4.2.4 on page 12).

[16] Create SGF (§ 4.2.5 on page 12).

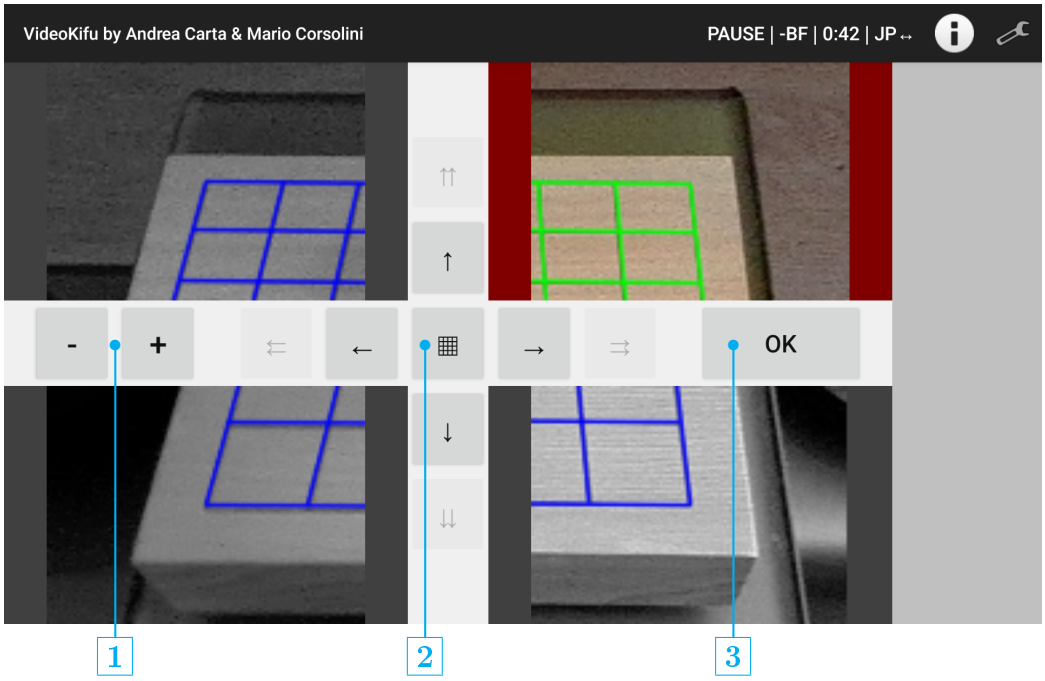


Figure 4: grid setting.

When no corners are selected (all four images are grey and the grid is blue) double arrow buttons shift the whole grid by one row/column.

When one (out of four) button is selected (its image is in colour and the grid is green), single arrow buttons shift (by half a pixel) only the nearest corner.

1 Increases/decreases the four images' zoom.

2 Sets a $9 \times 9$ or $13 \times 13$ or $19 \times 19$ grid (works only before the start of game).

3 Upholds the grid as currently set.

#### 4.2.10 Close VideoKifu

Closes the app.

Starting again the app will make it possible to record a new game.

## 4.3 User interface's other controls

### 4.3.1 Status bar

While the app is running, some useful infos about the current processings are shown on the status bar (3 in figure 1 on page 9 or 2 in figure 2 on page 10). From left to right:

- The symbol "↻" if the device is not placed in "landscape" mode.
- Number of frames processed for second, if the recording is running.
- The letters "SBF" if, respectively: the **S**GF file has been saved and it is still current,[17] **B**roadcast online is active and **F**rames' saving is active. If the SGF file has not been saved or some of the other tasks are disabled the simbol "-" will appear in place of the corresponding letters.
- How much time has passed since the app has started or, once upheld the grid position, since the start of the recording.
- The kind of goban that has been detected[18], placed either in parallel (↕) or perpendicularly (↔) to the camera's optical axis or, once upheld the grid position, the number of frames processed.

### 4.3.2 Processed frame

In every moment, by means of pinching or touching twice the frame taking up most of the main window (2 in figure 1 on page 9 or 1 in figure 2 on page 10), the displayed image may be expanded[19] or shrunken.

[17] If it is obsolete the simbol "∂" will be shown instead of "S".
[18] Square ([+]), Chinese (CN), Japanese (JP) or XL.
[19] Much useful if the device's screen is small.

Either during the recording or when the game ends, touching an intersection will make appear, in the lower part of the window, a short message with the corresponding coordinates, and, had some stones been played there, the pertinent moves.

Besides, either during the recording or when the game ends, a long touch over an intersection will open a window that will allow the user to manually edit any move detected there, as shown in Moves' manual editing (§ 4.3.4 on page 17).

### 4.3.3 Moves' list

Moves that are detected by the app are shown in a list on the right side of the main window ( 3 in figure 2 on page 10). When touching a move in the list the corresponding intersection will be highlighted briefly in the Processed frame (§ 4.3.2 on page 16).

To make it easier searching for errors and other less important issues that could arise during the recording (as an handicap wrongly set or a 3–4 moves' sequence to be checked), the app inserts some warnings in the moves' list, with a different colour depending on the situation. Touching such a move will let a message appear, with the description of the corresponding warnings. The warnings could be dealt with by manually editing the move by means of a long touch on the move itself: the Moves' manual editing (§ 4.3.4 on page 17) window will open, and only the options related to that specific move will be present.

### 4.3.4 Moves' manual editing

The app should correct by itself most errors related to moves' detection, such as false positives or stones not detected at all (because of unfavorable lighting or because played far from the intersection). In any case the user too may correct manually such errors: actually it's best to do that as soon as possible, in order to prevent the wrong position to disrupt the subsequent recording.

According to what has been touched by the user (refer to figure 2 on page 10), the following actions will be available:

**Empty intersection on the goban** 1

This makes it possible to add a move (or a Pass) on the intersection touched;[20] as well as an handicap stone; it's also possible to relocate a previous move on this intersection.

**Stone on the goban** 1

The touched stone could be removed[20] or replaced with a Pass. If several moves have been played on the intersection, the one concerned will be the last one, but another one could be chosen from a list.

Besides editing a move it will be possible to manage the related warnings. If some are present they could be deleted, otherwise the move could be set to "verified": this way the app won't ever be able to remove it, should even believe the move to be a false positive.

**Move in the list** 3

Available actions and advice's management are the same as the previous case, just without the choice among different moves played on the same intersection.

# 5. How to uninstall

Apply the ordinary procedure for uninstalling apps established by the version of Android in use and remove, if they have been created, the local copies of the files saved by the app. That will completely uninstall VideoKifu. :-(

# 6. Legal policies

Graphic computations are carried out by the library OpenCV, distributed under the Apache-2.0 licence.

Zoom functionality of frames displayed in the main window is supplied by TouchImageView, distributed under the MIT licence.

[20] If other moves after the one just added are present, they will be inverted in pair, in order to maintain their colour. The last detected move could possibly be deleted.

The viewer/editor used for online broadcast of the games is maxiGos, distributed under the BSD-3-Clause licence.

Data interchange between "VideoKifu for Android" and FTP servers is managed by the library ftp4j, distributed under the LGPL-2.1 licence.

## 6.1 Personal data and ads

"VideoKifu for Android" does not collect any personal data. Neither analytics nor advertising tools are present in "VideoKifu for Android".

"VideoKifu for Android" saves videos shot by means of the main video camera of the device where it runs: the user is fully responsible for all information contained within. In most cases, those videos are contained within the device where "VideoKifu for Android" runs and not accessible externally. Enabling external access requires manual steps and should be done at the user own risk. The videos stored by "VideoKifu for Android" may be deleted by means of any standard file managing tool.

## 6.2 Disclaimer of warranty and limitation of liability

THERE IS NO WARRANTY FOR "VIDEOKIFU FOR ANDROID", TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING, THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE "VIDEOKIFU FOR ANDROID" "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF "VIDEOKIFU FOR ANDROID" IS WITH THE USER. SHOULD "VIDEOKIFU FOR ANDROID" PROVE DEFECTIVE, THE USER ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR

ANY OTHER PARTY WHO CONVEYS "VIDEOKIFU FOR ANDROID", BE LIABLE TO THE USER FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE "VIDEOKIFU FOR ANDROID" (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY THE USER OR THIRD PARTIES OR A FAILURE OF "VIDEOKIFU FOR ANDROID" TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with "VideoKifu for Android", unless a warranty or assumption of liability accompanies a copy of "VideoKifu for Android" in return for a fee.

# 7. Feedback, suggestions, bug-reports

Any comments, suggestions and (most of all) bug-reports are welcome. Please use the eMail address `andrea.carta@mclink.it` or `mario@corsolini.net`

While submitting a bug it is advisable to specify "VideoKifu for Android" in the subject field and to report the device used as well as the version of both the app[21] and Android. It is also advisable to describe and attach anything useful to reproduce and analyse the problem.

VideoKifu is a multilingual app: contact us if you are willing to translate it into another language!

[21] Check whether it is the most updated one, on Google Play. In case it is not, please update the app and check whether the issue is still occurring.

# 8. Release history

- Version **1.3.1** — April 30$^{th}$, 2025
  - Bug fixed: VideoKifu crashed on Android 16 when rotating the device.
  - Bug fixed: all the windows were poorly displayed on Android 16.
  - Minor cosmetic changes and bug fixes.

- Version **1.3.0** — March 14$^{th}$, 2025
  - Added "Video source" in "App settings".
  - Improved visualisation of the frames.
  - Bug fixed: change of alias during online broadcast was mismanaged.
  - Removed "Visualisation" from "App settings".
  - Minor cosmetic changes and bug fixes.

- Version **1.2.0** — December 21$^{st}$, 2024
  - Added a final report on produced files.
  - Added menu item "Show report".
  - Added "Save time-lapse frames of the game" in "App settings".
  - Bug fixed: moves' coordinates were not rotated and players' colours were not translated in the comments of the SGF file.
  - Removed "Save video" from "App settings".
  - Removed "Save frames of recognised moves" from "App settings".
  - Minor cosmetic changes and bug fixes.

- Version **1.1.0** — October 31$^{st}$, 2024
  - Added confirmation or selection at the end of the game of the actual last move.
  - Added automatic setting of the point of view for online broadcast.
  - Added automatic per-app language support in Android 13 or later (not to be used while recording a game as switching language resets the app).
  - Added German translation of the app.

  - Improved the aspect of the online viewer/editor, added automatic refresh and live frames during broadcast, info about the game when available, marks on the stones potentially wrongly recognised.
  - Removed "Send SGF file only" from "App settings".
  - Minor cosmetic changes and bug fixes.

- Version **1.0.0** — March 20th 2024
  - First version publicly released.

# 9. Acknowledgements

Thanks to the authors of OpenCV,[22] TouchImageView,[23] maxiGos[24] and ftp4j[25] for providing their useful libraries.

Thanks to Taobioni for the German translation of the app.

Moreover, thanks to all the people who helped with ideas and valuable suggestions, particularly: Dani Ferrari for his advice on the stone recognition algorithm, Hyun-Soo Park[26] (and coworkers) for his works on a static score-counting algorithm, as well as the beta testers!

And, obviously, thanks to Laura and EmmeTi!!

Well, that's all about it, happy playing!!!

[22] https://opencv.org/
[23] https://github.com/MikeOrtiz/TouchImageView/
[24] https://www.jeudego.org/maxiGos/
[25] https://www.sauronsoftware.it/projects/ftp4j/
[26] Department of Computer Information Technology, Kyungdong College of Techno-Information, 224-1, Buho, Hayang, Kyungpook, Republic of Korea.