# DartScore

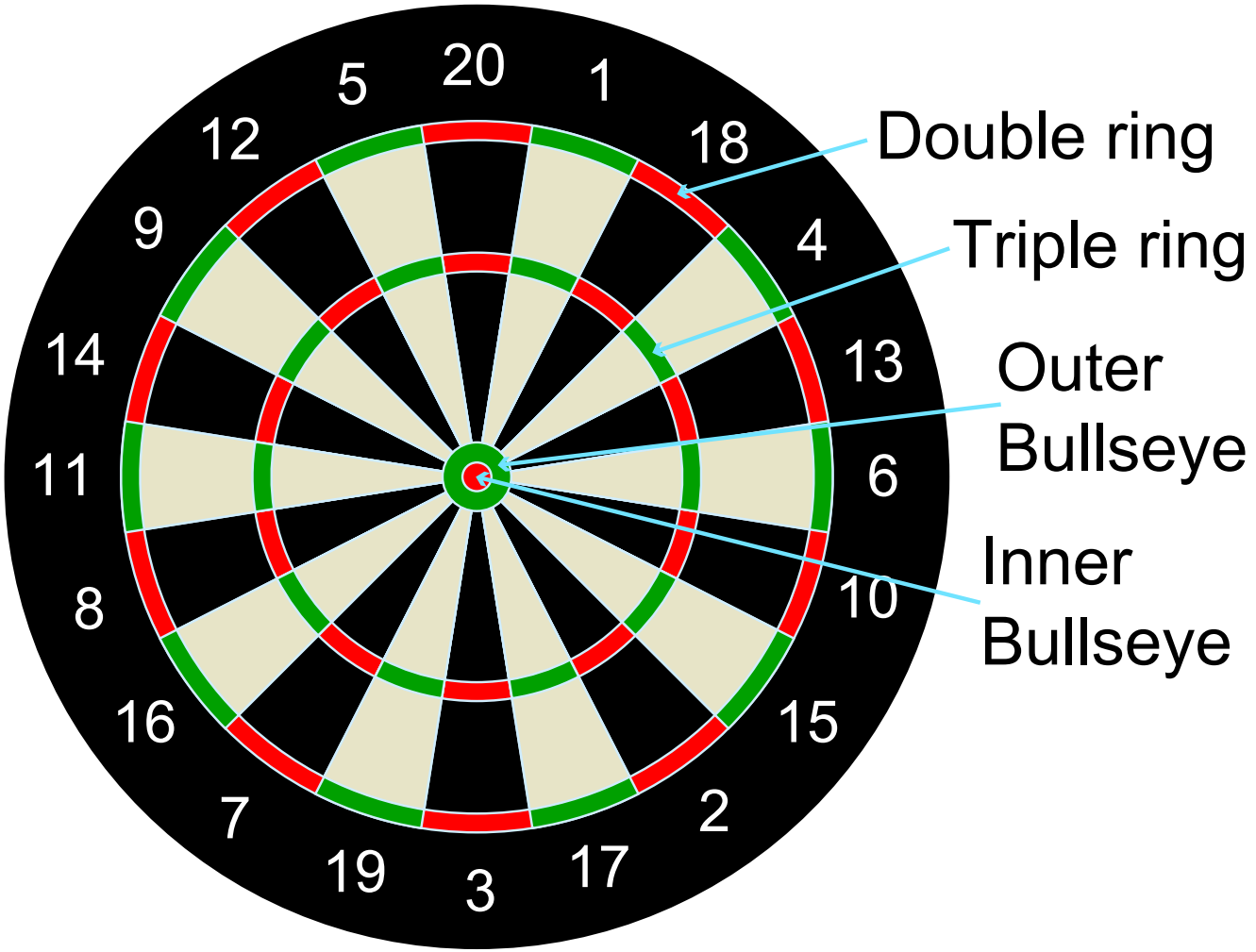
# https://en.wikipedia.org/wiki/Darts

Darts is a sport in which small missiles are thrown at a circular target ("dartboard") fixed to a wall.

Though various boards and rules have been used in the past, the term "darts" usually now refers to a standardised game involving a specific board design and set of rules.

As well as being a professional competitive game, darts **is a traditional pub game**, commonly played in the United Kingdom and the Republic of Ireland, across the Commonwealth, the Netherlands, Belgium, Germany, the Scandinavian countries, the United States and elsewhere.

# How to count the scores



# How to play (example)

- A game of darts is generally contested between two players, who take turns. The most common objective is to reduce a fixed score, commonly 301 or 501, to zero ("checking out"). The final dart must land in either the bullseye or a double segment in order to win. It is not necessary that all three darts are thrown on the final turn, the game can be finished on any of the three darts.

- "Killer" is a 'knock-out' game for two or more players (at its best at 4–6 players). Initially, each player throws a dart at the board with their non-dominant hand to obtain their 'number'. No two players can have the same number. Once everyone has a number, each player takes it in turn to get their number five times with their three darts (doubles count twice, and triples three times). Once a person has reached 5, they become a 'killer'. This means they can aim for other peoples numbers, taking a point off for each time they hit (doubles x2, triples x3). If a person gets to zero they are out. A killer can aim for anyone's numbers, even another killer's. Players cannot get more than 5 points. The winner is 'the last man standing'.

# DartScore

A project turned on and off during for *at least* five years...

It started with a dart-board as a gift from my wife and a bit later I came across some interesting python-code using opencv...

Or actually, it probably started 35+ years ago at some pub in Luleå...


# What is the purpose of DartScore?

- Count scores in darts (with image recognition)
- Have a setup that is rather easy for others to follow and use
  - Structured code
  - Standard components
- **For fun,** not for competition


=> Using my favourite stuff:

  - RaspberryPi, Python and some extra electronics

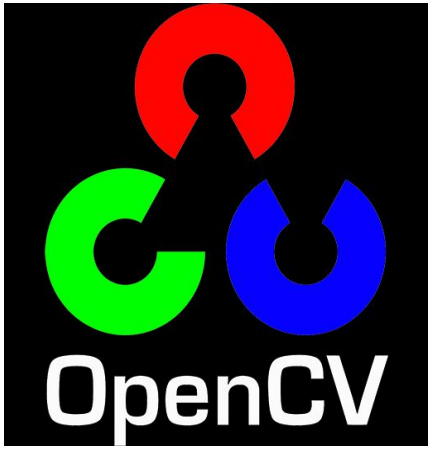
# My setup


* DartScore *
Player 1
Player 2
Dart 1: 0
Dart 2: 0
Dart 3: 0
Total: 0
OFFICIAL TOURNAMENT

# The software

- Python 3.x, OpenCv, PyGame
- Separate parts/modules for
  - The Gamestate(s)
    - Mount, calibrate, play...
  - The FrontEnd
    - Each gamestate has its own FrontEnd
  - The DartScoreEngine, takes care of
    - Calibration algorithms
    - DartDetection
    - DartHit coordinates
    - Board geometry
  -



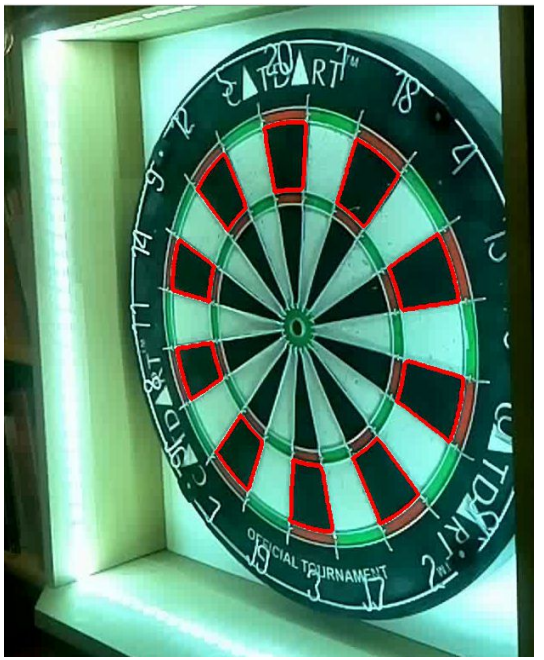


# My tactics for the project

- Good lighting conditions!
- Camera from upper right (as a start)
- Calibrate and transform the picture of the board to 'the perfect view'
  - Find bulls eye
  - Find 4 sector 'corners'
  - Transform...
- Find the coordinates for the tip of the dart
- Use the coordinates to calculate the scores
- Piece of cake (not!)

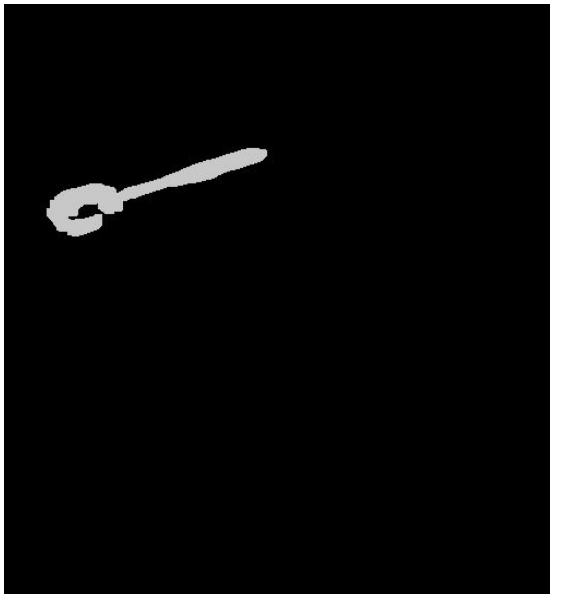
# The hard parts 1

- Calibration of the board vs the image stream
  - Calculate the transform matrix between the skewded image and the perfect board
  - Used methods:
    - HoughesLinesP
    - FindContours
    - Shape-detection (corners)
    - GetPerspectiveTransform



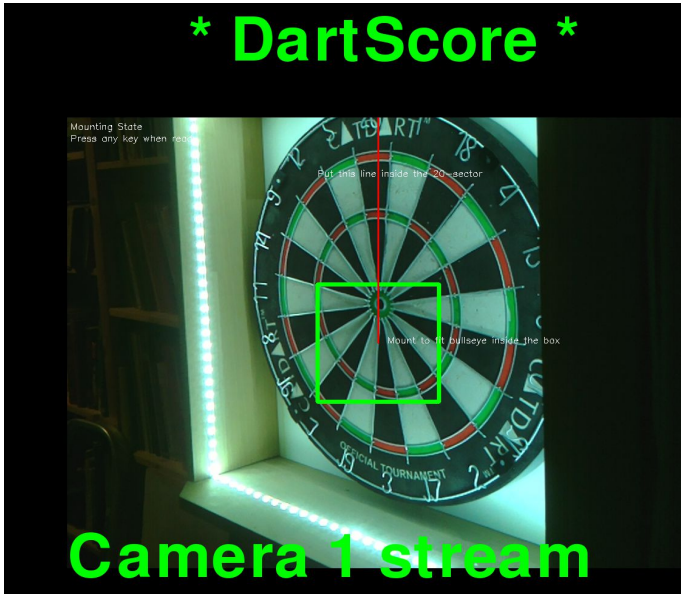
# The hard parts 2

- Finding the darts..
- Used methods:
  - Absdiff(frame1, frame2)
  - Masking
  - FindContours
- Detecting the 'tip'

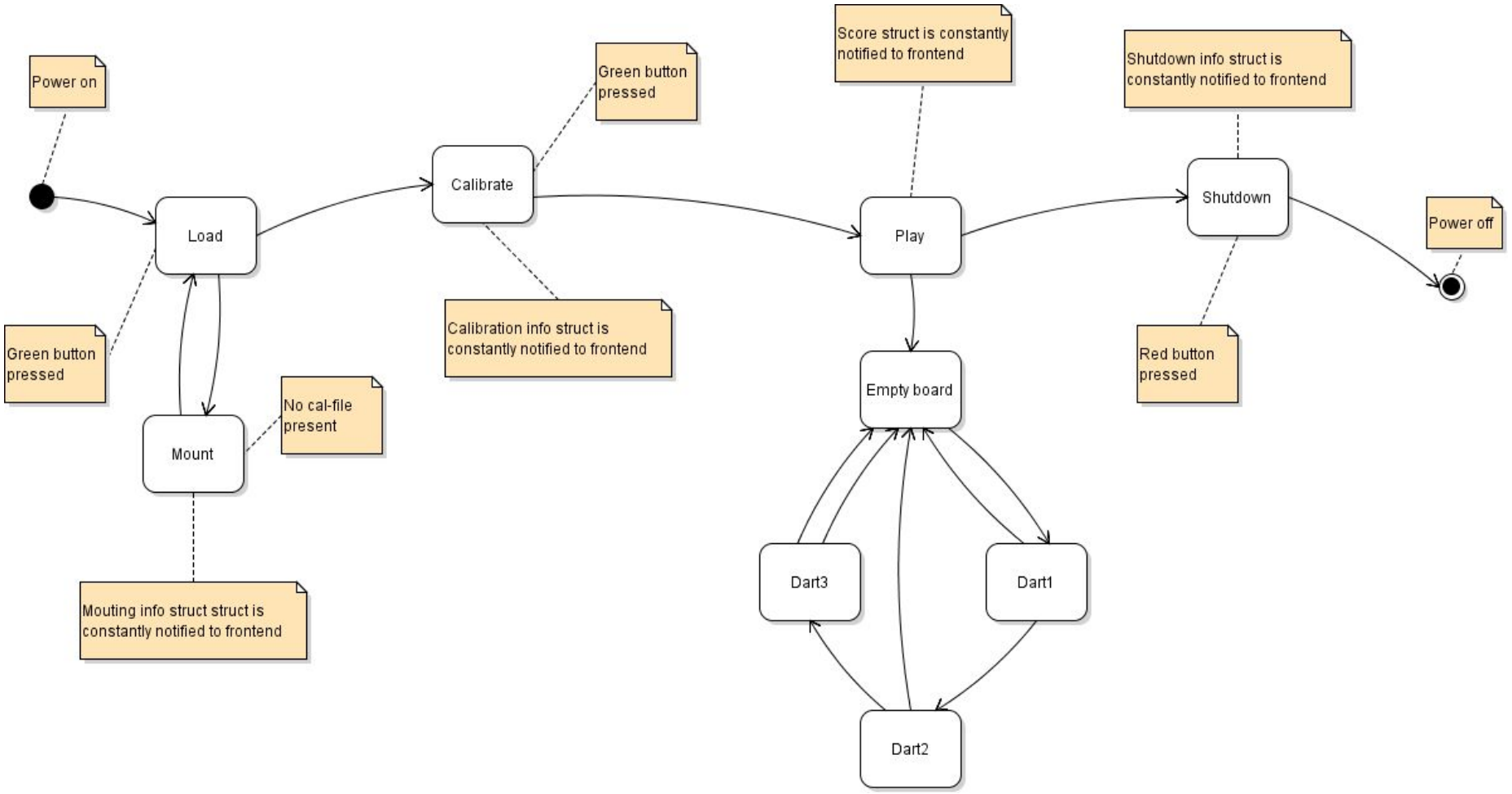


# The hard parts 3

- State machines on several levels:
  - Cam mounting state
  - Calibration state
    - BullsEye
    - Sectors and calibration points
    - Transform matrix
  - Game-states
    - Board empty
    - Dart 1, 2 | 3
    - Board stabilized but not empty...



# Game states


Power on
Load
Green button pressed
Mount
No cal-file present
Mouting info struct struct is constantly notified to frontend
Calibrate
Green button pressed
Calibration info struct is constantly notified to frontend
Score struct is constantly notified to frontend
Play
Empty board
Dart3
Dart1
Dart2
Shutdown
Shutdown info struct is constantly notified to frontend
Red button pressed
Power off

# My hardware

- Dartboard
- Lightbox with led-strip
- RaspberryPi 4 2GB (main unit)
  - Running standard Rasbian
  - (Fan, buttons, temp. sensor...)
- RaspberryPi 3 with RaspiCam (netcam)
  - Running MotioneyeOs in fast networked cam mode
- Monitor
- Cables and power-supply

# DartScore MVP

Currently the project is in 'MVP' (minimum viable product) status wich means that it has a minimum of functionallity but can be used for real. The supported functionallity is:

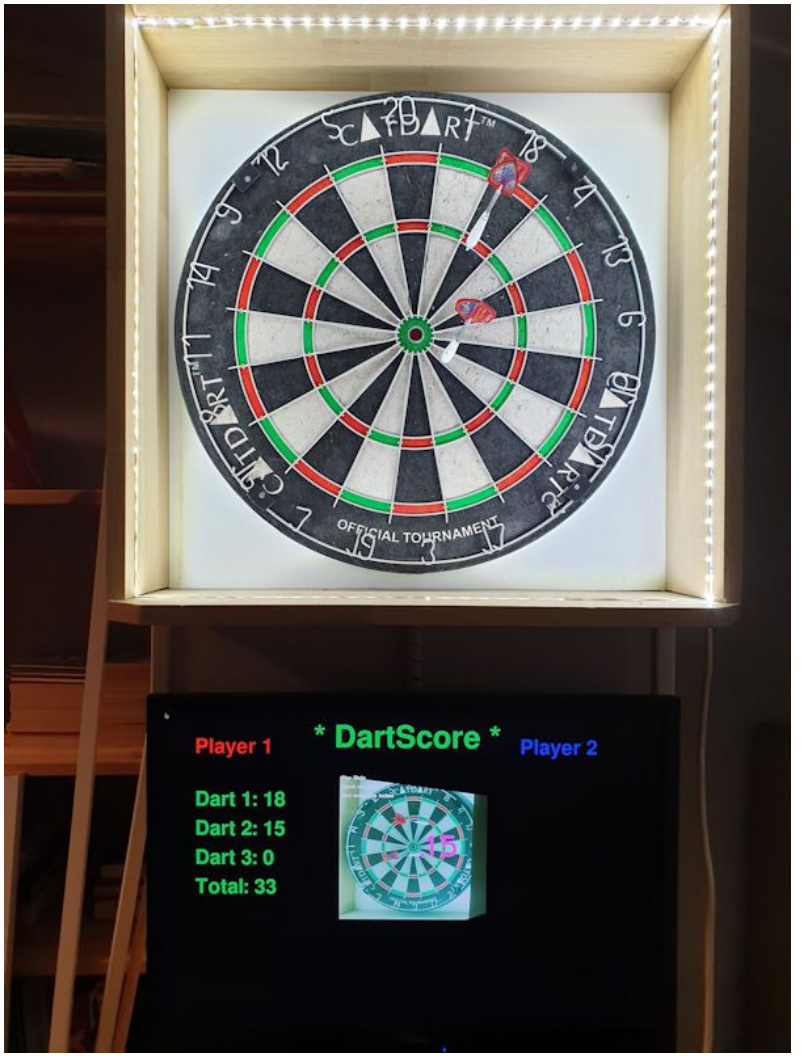
Mounting state to help setting up the cam

Calibration state to calibrate and transform the image stream from one cam and saving/reading calibration matrix from file

Playstate supporting one player and a simple algorithm for counting scores for each set of darts but not very accurate right now.

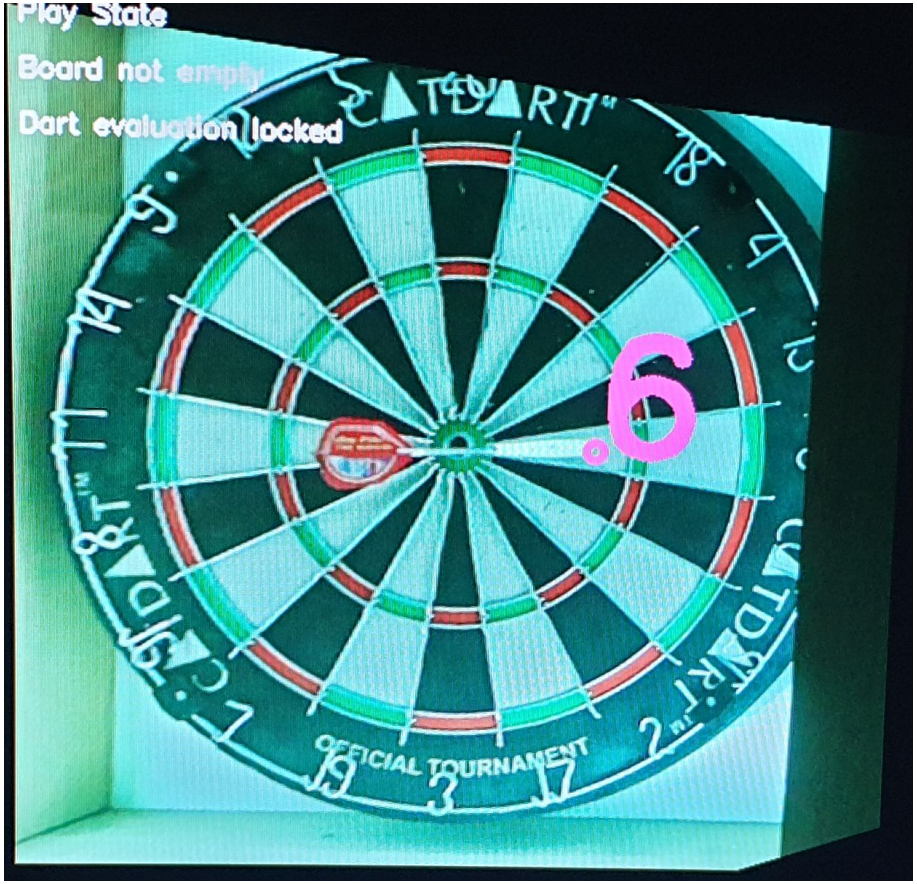
All states has a hardcoded gui that is presented on a screen with 1680x1050 resolution

But it works (sort of…) !



# The result


OFFICIAL TOURNAMENT


Play State
Board not empty
Dart evaluation locked
6
OFFICIAL TOURNAMENT

# Improvements, in future updates

Performace optimisation (main issue right now)

A better algorithm for counting the scores (main issue right now)

Support for IO: buttons, leds, temperature-monitor and fans

Configurable gui (To make it easier for others to use)

Support for real and configurable gametypes like 301, 501 etc (To be able to use it for real)

Scoreboard in the cloud (Better incitament for others to use)

Support for 2 players (To be able to use it for real)

Support for 2 cams (To further improve the score calculation)

Saving game stats and be able to continue a game (Nice to have)

… more to come…

# Movie!

"C:\Users\par\OneDrive\Documents\GitHub\DartScore\Docs\20191217_151832.mp4"