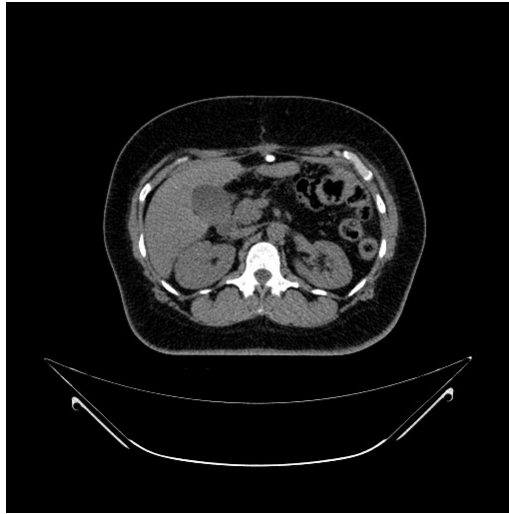


Dicom Detect with YOLOv7

Luiz Castiglioni

30/09/2022



Trabalho de conclusão de curso do aluno Luiz Castiglioni no curso do MBA - Data Science and ANalytics da USP / ESALQ. Trabalho desenvolvido em parceria com o departamento de AI da Escola Paulista de Medicina UNIFESP.

Data info Tomographic images and Metadata of Medical Exams in DICOM Standards Format

- total files: 165371
- total size: 89.9GB
- format: “.dcm”

Links

- Dicom Medicine Standards <https://www.dicomstandard.org/ai>

- YOLO <https://pjreddie.com/darknet/yolo/>
- Github https://github.com/luizcast/DCM_TCC

```
list.files(include.dirs = TRUE)
```

Arquivos do projeto

```
## [1] "1PRIMEIRO_SCRIPT.R"      "2SEGUNDO_SCRIPT.R"
## [3] "3TERCEIRO_SCRIPT.R"     "4QUARTO_SCRIPT.R"
## [5] "5QUINTO_SCRIPT.R"       "Anot_IMGS.R"
## [7] "config_nvidia.png"      "dcm_tools.Rproj"
## [9] "IMGs"                   "keras.R"
## [11] "markdown_dicom_yolo_files" "markdown_dicom_yolo.docx"
## [13] "markdown_dicom_yolo.html" "markdown_dicom_yolo.pdf"
## [15] "markdown_dicom_yolo.Rmd"  "rascunho.R"
## [17] "README.md"               "TABLES"
## [19] "yolo_dicom_v1.pt"
```

Pacotes utilizados no projeto

- oro.dicom - <https://cran.r-project.org/web/packages/oro.dicom/index.html>
- dcmktk - <https://dicom.offis.de/dcmktk.php.en>
- tidyverse - <https://www.tidyverse.org/>
- tensorflow - <https://www.tensorflow.org/>
- keras - <https://keras.io/>

```
pacotes <- c("keras", "tensorflow", "tidyverse", "oro.dicom", "dcmktk", "readxl")
if(sum(as.numeric(!pacotes %in% installed.packages())) != 0){
  instalador <- pacotes[!pacotes %in% installed.packages()]
  for(i in 1:length(instalador)) {
    install.packages(instalador, dependencies = T)
    break()}
  sapply(pacotes, require, character = T)
} else {
  sapply(pacotes, require, character = T)
}
```

```
## Loading required package: keras
```

```
## Loading required package: tensorflow
```

```
## Loading required package: tidyverse
```

```
## -- Attaching packages ----- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v purrr    0.3.4
## v tibble  3.1.8      v dplyr   1.0.9
## v tidyr   1.2.0      v stringr 1.4.0
## v readr   2.1.2      v forcats 0.5.1
## -- Conflicts ----- tidyverse_conflicts() --
```

```
## x dplyr::filter() masks stats::filter()
## x dplyr::lag() masks stats::lag()
## Loading required package: oro.dicom
##
## oro.dicom 0.5.3
##
## Loading required package: dcmtk
##
## Loading required package: readxl

##      keras tensorflow tidyverse oro.dicom      dcmtk      readxl
##      TRUE      TRUE      TRUE      TRUE      TRUE      TRUE
```

```
tf_gpu_configured()
```

```
## Loaded Tensorflow version 2.9.2
```

```
## TensorFlow built with CUDA: TRUE
```

```
## GPU device name: /device:GPU:0
```

```
## [1] TRUE
```

```
tf_config()
```

```
## TensorFlow v2.9.2 (~/.local/share/r-miniconda/envs/r-reticulate/lib/python3.8/site-packages/tensorflow)
## Python v3.8 (~/.local/share/r-miniconda/envs/r-reticulate/bin/python)
```

Configurações da Máquina

```
neofetch --stdout
```

```
## luiz@box
## -----
## OS: Linux Mint 21 x86_64
## Host: Dell G15 5515 1.7.1
## Kernel: 5.15.0-48-generic
## Uptime: 5 hours, 40 mins
## Packages: 3375 (dpkg), 9 (flatpak), 11 (snap)
## Shell: bash 5.1.16
## Resolution: 1920x1080
## DE: Cinnamon
## WM: Muttter
## WM Theme: Mint-Y
## Theme: Mint-Y-Dark-Purple [GTK2/3]
## Icons: Mint-Y-Dark-Blue [GTK2/3]
## Terminal: R
## CPU: AMD Ryzen 7 5800H with Radeon Graphics (16) @ 4.463GHz
## GPU: NVIDIA GeForce RTX 3060 Mobile / Max-Q
## GPU: AMD ATI 06:00.0 Cezanne
## Memory: 6086MiB / 15342MiB
```

```
nvidia-smi
```

```
## Fri Sep 30 21:45:32 2022
## +-----+
## | NVIDIA-SMI 515.65.01    Driver Version: 515.65.01    CUDA Version: 11.7    |
## |-----+-----+-----+
## | GPU   Name               Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
## | Fan  Temp  Perf  Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
## |                               |                  |           MIG M.     |
## |=====+=====+=====+
## |    0  NVIDIA GeForce ...   Off      | 00000000:01:00.0 Off |             N/A     |
## | N/A   54C    P0      21W /   N/A   |  891MiB /  6144MiB |      21%      Default |
## |                               |                  |           N/A     |
## |-----+-----+-----+
##
## +-----+
## | Processes:                                |
## | GPU   GI    CI          PID    Type    Process name                        GPU Memory |
## |       ID    ID                                   Usage      |
## |=====+=====+=====+
## |    0   N/A   N/A         1292     G   /usr/lib/xorg/Xorg                    195MiB |
## |    0   N/A   N/A         2548     G   cinnamon                             66MiB |
## |    0   N/A   N/A         3267     G   /usr/lib/rstudio/bin/rstudio          66MiB |
## |    0   N/A   N/A         3320     C   .../lib/rstudio/bin/rsession         273MiB |
## |    0   N/A   N/A        27357     G   ...090296402576475813,131072         11MiB |
## |    0   N/A   N/A        67801     C   /usr/lib/R/bin/exec/R                 273MiB |
## |-----+-----+-----+
```

Reconhecendo Padrões em imagens de Tomografia de Abdómen no formato DICOM.

Foram selecionados 200 exames de tomografia de abdômen de 200 pacientes diferentes. Desses 200 exames foram geradas anotações pelo grupo de médicos e residentes da UNIFESP sobre a presença do órgão vesícula ou a presença do clipe cirúrgico em casos de pacientes que tiveram a vesícula retirada. Surgiram 4 classes:

- Pacientes com Vesícula Normal
- Pacientes com Vesícula Hipodistendida
- Pacientes sem Vesícula e com Clipe Cirúrgico
- Pacientes sem Vesícula e sem Clipe Cirúrgico

Destas anotações surgiu a tabela **annot_clean**

```
read_excel("~/Desktop/USP/TCC/projeto_TCC/TABLEs/annot_clean.xlsx") %>% as_tibble() -> annot_clean
annot_clean
```

```
## # A tibble: 200 x 10
##   Acession Num~1 Patie~2 Corte~3 Corte~4 Hipod~5 Densi~6 Sem c~7 Arter~8 Portal
##   <dbl>    <dbl> <chr>   <chr>   <chr>   <chr>   <dbl>   <dbl>   <dbl>
## 1      5499993 3861927 Não se~ Não se~ S      Não se~     1       1       1
## 2      5493468 823083  Não se~ Não se~ C      Não se~     1       1       1
```

```
## 3      5480343 3586593 Não se~ Não se~ S      Não se~      0      1      1
## 4      5481425 2863787 Não se~ Não se~ C      Não se~      0      0      1
## 5      5480051 3880103 Não se~ Não se~ C      Não se~      0      0      1
## 6      5459110 3825619 Não se~ Não se~ S      Não se~      0      0      1
## 7      5465667 3132902 Não se~ Não se~ C      Não se~      1      1      1
## 8      5466142 3872936 Não se~ Não se~ C      Não se~      1      0      1
## 9      5435830 3227017 Não se~ Não se~ C      Não se~      1      1      1
## 10     5433748 870140  Não se~ Não se~ S      Não se~      0      0      1
## # ... with 190 more rows, 1 more variable: Equilibrio <dbl>, and abbreviated
## #   variable names 1: 'Acession Number', 2: 'Patient ID',
## #   3: 'Corte INICIAL vesícula', 4: 'Corte FINAL vesícula',
## #   5: 'Hipodistendida/Vesícula normal/Clipe/sem clipe (HVSC)',
## #   6: 'Densidade não habitual (sim=1, não=0)',
## #   7: 'Sem contraste (sim = 1; não = 0)', 8: Arterial
## # i Use 'print(n = ...)' to see more rows, and 'colnames()' to see all variable names
```

Preparando a Base Annot_Clean

```
gsub(pattern = "Não se aplica", replacement = "NA",
      annot_clean$`Densidade não habitual (sim=1, não=0)` ->
      annot_clean$`Densidade não habitual (sim=1, não=0)`
gsub(pattern = "Não se aplica", replacement = "NA",
      annot_clean$`Corte INICIAL vesícula` ->
      annot_clean$`Corte INICIAL vesícula`
gsub(pattern = "Não se aplica", replacement = "NA",
      annot_clean$`Corte FINAL vesícula` ->
      annot_clean$`Corte FINAL vesícula`
annot_clean$`Hipodistendida/Vesícula normal/Clipe/sem clipe (HVSC)` <-
  as.factor(annot_clean$`Hipodistendida/Vesícula normal/Clipe/sem clipe (HVSC)`)
annot_clean$`Corte FINAL vesícula` <-
  as.numeric(annot_clean$`Corte FINAL vesícula`)
annot_clean$`Corte INICIAL vesícula` <-
  as.numeric(annot_clean$`Corte INICIAL vesícula`)
annot_clean$`Densidade não habitual (sim=1, não=0)` <-
  as.numeric(annot_clean$`Densidade não habitual (sim=1, não=0)`)
annot_clean
```

```
## # A tibble: 200 x 10
##   Accesion Num~1 Patie~2 Corte~3 Corte~4 Hipod~5 Densi~6 Sem c~7 Arter~8 Portal
##           <dbl>   <dbl>   <dbl>   <dbl> <fct>         <dbl>   <dbl>   <dbl>   <dbl>
## 1      5499993 3861927      NA      NA S           NA       1       1       1
## 2      5493468 823083      NA      NA C           NA       1       1       1
## 3      5480343 3586593      NA      NA S           NA       0       1       1
## 4      5481425 2863787      NA      NA C           NA       0       0       1
## 5      5480051 3880103      NA      NA C           NA       0       0       1
## 6      5459110 3825619      NA      NA S           NA       0       0       1
## 7      5465667 3132902      NA      NA C           NA       1       1       1
## 8      5466142 3872936      NA      NA C           NA       1       0       1
## 9      5435830 3227017      NA      NA C           NA       1       1       1
## 10     5433748 870140      NA      NA S           NA       0       0       1
## # ... with 190 more rows, 1 more variable: Equilibrio <dbl>, and abbreviated
## #   variable names 1: 'Acession Number', 2: 'Patient ID',
```

```
## # 3: 'Corte INICIAL vesícula', 4: 'Corte FINAL vesícula',
## # 5: 'Hipodistendida/Vesícula normal/Clipe/sem clipe (HVSC)',
## # 6: 'Densidade não habitual (sim=1, não=0)',
## # 7: 'Sem contraste (sim = 1; não = 0)', 8: Arterial
## # i Use 'print(n = ...)' to see more rows, and 'colnames()' to see all variable names
```

```
colnames(annot_clean)
```

```
## [1] "Acession Number"
## [2] "Patient ID"
## [3] "Corte INICIAL vesícula"
## [4] "Corte FINAL vesícula"
## [5] "Hipodistendida/Vesícula normal/Clipe/sem clipe (HVSC)"
## [6] "Densidade não habitual (sim=1, não=0)"
## [7] "Sem contraste (sim = 1; não = 0)"
## [8] "Arterial"
## [9] "Portal"
## [10] "Equilibrio"
```

Criando Dataset

```
path = "/home/luiz/Desktop/DATA-SET_TESTE/"
list.files(path) %>% as_tibble() -> array_arquivos_dicom_tratados
separate(array_arquivos_dicom_tratados, col = value, sep = "_", into = c("PatientID", "SerieNumber", "InstanceNumber"))
index_dicom$InstanceNumber %>% str_sub(end=-5) -> index_dicom$InstanceNumber
bind_cols(array_arquivos_dicom_tratados, index_dicom) -> index_dicom
names(index_dicom) <- c("FileName", "PatientID", "SerieNumber", "InstanceNumber")

index_dicom$PatientID <- as.numeric(index_dicom$PatientID)
index_dicom$SerieNumber = as.numeric(index_dicom$SerieNumber)
index_dicom$InstanceNumber = as.numeric(index_dicom$InstanceNumber)

#join com annot_clean
index_dicom %>% group_by(PatientID) %>%
  left_join(annot_clean, by = c("PatientID" = "Patient ID")) %>%
  ungroup() -> index_dicom
index_dicom$FilePath <- paste(path, index_dicom$FileName, sep='')
index_dicom %>% dplyr::select(1:9) -> base_tratada
base_tratada %>% dplyr::select(-5) -> base_tratada
base_tratada$FilePath <- paste(path, base_tratada$FileName, sep='')
base_tratada
```

```
## # A tibble: 165,794 x 9
##   FileName      Patie~1 Serie~2 Insta~3 Corte~4 Corte~5 Hipod~6 Densi~7 FileP~8
##   <chr>         <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <fct>   <dbl> <chr>
## 1 0000121029_2~ 121029     2       1     NA     NA S     NA /home/~
## 2 0000121029_2~ 121029     2      10     NA     NA S     NA /home/~
## 3 0000121029_2~ 121029     2     100     NA     NA S     NA /home/~
## 4 0000121029_2~ 121029     2     101     NA     NA S     NA /home/~
## 5 0000121029_2~ 121029     2     102     NA     NA S     NA /home/~
## 6 0000121029_2~ 121029     2     103     NA     NA S     NA /home/~
```

```

## 7 0000121029_2~ 121029      2      104      NA      NA S      NA /home/~
## 8 0000121029_2~ 121029      2      105      NA      NA S      NA /home/~
## 9 0000121029_2~ 121029      2      106      NA      NA S      NA /home/~
## 10 0000121029_2~ 121029      2      107      NA      NA S      NA /home/~
## # ... with 165,784 more rows, and abbreviated variable names 1: PatientID,
## # 2: SerieNumber, 3: InstanceNumber, 4: 'Corte INICIAL vesícula',
## # 5: 'Corte FINAL vesícula',
## # 6: 'Hipodistendida/Vesícula normal/Clipe/sem clipe (HVSC)',
## # 7: 'Densidade não habitual (sim=1, não=0)', 8: FilePath
## # i Use 'print(n = ...)' to see more rows

```