sensor_msgs/CameraInfo Message

File: sensor_msgs/CameraInfo.msg

Raw Message Definition

```
# This message defines meta information for a camera. It should be in a
# camera namespace on topic "camera_info" and accompanied by up to five
# image topics named:
#
#   image_raw - raw data from the camera driver, possibly Bayer encoded
#   image            - monochrome, distorted
#   image_color      - color, distorted
#   image_rect       - monochrome, rectified
#   image_rect_color - color, rectified
#
# The image_pipeline contains packages (image_proc, stereo_image_proc)
# for producing the four processed image topics from image_raw and
# camera_info. The meaning of the camera parameters are described in
# detail at http://www.ros.org/wiki/image_pipeline/CameraInfo.
#
# The image_geometry package provides a user-friendly interface to
# common operations using this meta information. If you want to, e.g.,
# project a 3d point into image coordinates, we strongly recommend
# using image_geometry.
#
# If the camera is uncalibrated, the matrices D, K, R, P should be left
# zeroed out. In particular, clients may assume that K[0] == 0.0
# indicates an uncalibrated camera.

#######################################################################
#                     Image acquisition info                          #
#######################################################################
```

```
# Time of image acquisition, camera coordinate frame ID
Header header    # Header timestamp should be acquisition time of image
                 # Header frame_id should be optical frame of camera
                 # origin of frame should be optical center of camera
                 # +x should point to the right in the image
                 # +y should point down in the image
                 # +z should point into the plane of the image


##########################################################################
#                     Calibration Parameters                             #
##########################################################################
# These are fixed during camera calibration. Their values will be the #
# same in all messages until the camera is recalibrated. Note that    #
# self-calibrating systems may "recalibrate" frequently.              #
#                                                                     #
# The internal parameters can be used to warp a raw (distorted) image #
# to:                                                                 #
#   1. An undistorted image (requires D and K)                        #
#   2. A rectified image (requires D, K, R)                           #
# The projection matrix P projects 3D points into the rectified image.#
##########################################################################

# The image dimensions with which the camera was calibrated. Normally
# this will be the full camera resolution in pixels.
uint32 height
uint32 width

# The distortion model used. Supported models are listed in
# sensor_msgs/distortion_models.h. For most cameras, "plumb_bob" - a
# simple model of radial and tangential distortion - is sufficient.
string distortion_model

# The distortion parameters, size depending on the distortion model.
# For "plumb_bob", the 5 parameters are: (k1, k2, t1, t2, k3).
float64[] D

# Intrinsic camera matrix for the raw (distorted) images.
#     [fx  0 cx]
# K = [ 0 fy cy]
#     [ 0  0  1]
# Projects 3D points in the camera coordinate frame to 2D pixel
# coordinates using the focal lengths (fx, fy) and principal point
```

```
# (cx, cy).
float64[9]  K # 3x3 row-major matrix

# Rectification matrix (stereo cameras only)
# A rotation matrix aligning the camera coordinate system to the ideal
# stereo image plane so that epipolar lines in both stereo images are
# parallel.
float64[9]  R # 3x3 row-major matrix

# Projection/camera matrix
#     [fx'  0  cx' Tx]
# P = [ 0  fy' cy' Ty]
#     [ 0   0   1   0]
# By convention, this matrix specifies the intrinsic (camera) matrix
#  of the processed (rectified) image. That is, the left 3x3 portion
#  is the normal camera intrinsic matrix for the rectified image.
# It projects 3D points in the camera coordinate frame to 2D pixel
#  coordinates using the focal lengths (fx', fy') and principal point
#  (cx', cy') - these may differ from the values in K.
# For monocular cameras, Tx = Ty = 0. Normally, monocular cameras will
#  also have R = the identity and P[1:3,1:3] = K.
# For a stereo pair, the fourth column [Tx Ty 0]' is related to the
#  position of the optical center of the second camera in the first
#  camera's frame. We assume Tz = 0 so both cameras are in the same
#  stereo image plane. The first camera always has Tx = Ty = 0. For
#  the right (second) camera of a horizontal stereo pair, Ty = 0 and
#  Tx = -fx' * B, where B is the baseline between the cameras.
# Given a 3D point [X Y Z]', the projection (x, y) of the point onto
#  the rectified image is given by:
#  [u v w]' = P * [X Y Z 1]'
#         x = u / w
#         y = v / w
#  This holds for both images of a stereo pair.
float64[12] P # 3x4 row-major matrix


########################################################################
#                     Operational Parameters                           #
########################################################################
# These define the image region actually captured by the camera       #
# driver. Although they affect the geometry of the output image, they #
# may be changed freely without recalibrating the camera.             #
########################################################################
```

```
# Binning refers here to any camera setting which combines rectangular
#  neighborhoods of pixels into larger "super-pixels." It reduces the
#  resolution of the output image to
#  (width / binning_x) x (height / binning_y).
# The default values binning_x = binning_y = 0 is considered the same
#  as binning_x = binning_y = 1 (no subsampling).
uint32 binning_x
uint32 binning_y

# Region of interest (subwindow of full camera resolution), given in
#  full resolution (unbinned) image coordinates. A particular ROI
#  always denotes the same window of pixels on the camera sensor,
#  regardless of binning settings.
# The default setting of roi (all values 0) is considered the same as
#  full resolution (roi.width = width, roi.height = height).
RegionOfInterest roi
```

Compact Message Definition

```
std_msgs/Header header
uint32 height
uint32 width
string distortion_model
float64[] D
float64[9] K
float64[9] R
float64[12] P
uint32 binning_x
uint32 binning_y
sensor_msgs/RegionOfInterest roi
```

autogenerated on Sun, 09 Feb 2020 03:18:29