tf.extract_image_patches

```
extract_image_patches(
    images,
    ksizes,
    strides,
    rates,
    padding,
    name=None
)
```

Defined in `tensorflow/python/ops/gen_array_ops.py`.

See the guide: Tensor Transformations > Slicing and Joining

Extract `patches` from `images` and put them in the "depth" output dimension.

Args:

- `images`: A `Tensor`. Must be one of the following types: `float32`, `float64`, `int32`, `int64`, `uint8`, `int16`, `int8`, `uint16`, `half`. 4-D Tensor with shape `[batch, in_rows, in_cols, depth]`.
- `ksizes`: A list of `ints` that has length `>= 4`. The size of the sliding window for each dimension of `images`.
- `strides`: A list of `ints` that has length `>= 4`. 1-D of length 4. How far the centers of two consecutive patches are in the images. Must be: `[1, stride_rows, stride_cols, 1]`.
- `rates`: A list of `ints` that has length `>= 4`. 1-D of length 4. Must be: `[1, rate_rows, rate_cols, 1]`. This is the input stride, specifying how far two consecutive patch samples are in the input. Equivalent to extracting patches with `patch_sizes_eff = patch_sizes + (patch_sizes - 1) * (rates - 1)`, followed by subsampling them spatially by a factor of `rates`. This is equivalent to `rate` in dilated (a.k.a. Atrous) convolutions.
- `padding`: A `string` from: `"SAME", "VALID"`. The type of padding algorithm to use.

  We specify the size-related attributes as:

  `python ksizes = [1, ksize_rows, ksize_cols, 1] strides = [1, strides_rows, strides_cols, 1] rates = [1, rates_rows, rates_cols, 1]` * `name`: A name for the operation (optional).

Returns:

A `Tensor`. Has the same type as `images`. 4-D Tensor with shape `[batch, out_rows, out_cols, ksize_rows * ksize_cols * depth]` containing image patches with size `ksize_rows x ksize_cols x depth` vectorized in the "depth" dimension. Note `out_rows` and `out_cols` are the dimensions of the output patches.

Last updated November 2, 2017.