tf.spectral.irfft2d

```
irfft2d(
    input_tensor,
    fft_length=None,
    name=None
)
```

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

See the guide: Spectral Functions > Discrete Fourier Transforms

Inverse 2D real-valued fast Fourier transform.

Computes the inverse 2-dimensional discrete Fourier transform of a real-valued signal over the inner-most 2 dimensions of `input`.

The inner-most 2 dimensions of `input` are assumed to be the result of `RFFT2D`: The inner-most dimension contains the `fft_length / 2 + 1` unique components of the DFT of a real-valued signal. If `fft_length` is not provided, it is computed from the size of the inner-most 2 dimensions of `input`. If the FFT length used to compute `input` is odd, it should be provided since it cannot be inferred properly.

Along each axis `IRFFT2D` is computed on, if `fft_length` (or `fft_length / 2 + 1` for the inner-most dimension) is smaller than the corresponding dimension of `input`, the dimension is cropped. If it is larger, the dimension is padded with zeros.

Args:

- `input`: A `Tensor` of type `complex64`. A complex64 tensor.
- `fft_length`: A `Tensor` of type `int32`. An int32 tensor of shape [2]. The FFT length for each dimension.
- `name`: A name for the operation (optional).

Returns:

A `Tensor` of type `float32`. A float32 tensor of the same rank as `input`. The inner-most 2 dimensions of `input` are replaced with the `fft_length` samples of their inverse 2D Fourier transform.

numpy compatibility

Equivalent to np.fft.irfft2

Last updated November 2, 2017.