(a) default with fixed batch size

```
dataset = tf.data.Dataset.from_tensor_slices(filenames)
dataset = dataset.map(_parse_function)
dataset = dataset.batch(fixed_batch_size)
dataset = dataset.repeat()
iterator = dataset.make_one_shot_iterator()
next_element = iterator.get_next()

...
val = sess.run(outlist)
```

(b) dynamic batch sizing enabled

```
imageString = tf.placeholder(tf.string,name='imageString')
imagenstack = tf.stack(imageString)
batch_size_dynamic = tf.placeholder(tf.int64, shape=(),
                                    name='batch_size_dynamic')
dataset = tf.data.Dataset.from_tensor_slices(imagenstack)
dataset = dataset.map(_parse_function)
dataset = dataset.batch(batch_size_dynamic)
dataset = dataset.repeat()
iterator = dataset.make_initializable_iterator()
next  element = iterator.get  next()

...
sess.run(iterator.initializer, feed_dict=
        {batch_size_dynamic: newbatchSize, imageString: StringImage_2})
val = sess.run(outlist,feed_dict=
        {batch_size_dynamic: newbatchSize, imageString: StringImage_2})
```