

# گزارش پروژه نهایی نوروساینس

## ریحانه درفشی ۶۱۰۳۹۶۰۹۸

پیاده سازی یک مدل محاسباتی مبتنی بر شبکه های عصبی اسپایکی برای تشخیص اشیا

### توضیح کد:

مدل محاسباتی HMAX با استفاده از bindsnet پیاده سازی شده است.  
لایه های مدل تشکیل شده است از:

#### 1. input:

چهار سائز عکس متفاوت که برای هر سائز در چهار جهت متفاوت فیلتر gabor با سائز ثابت اعمال میشود، در مجموع ۱۶ لایه.

#### 2. S1:

به همین ترتیب ۱۶ لایه که بین هر لایه و لایه متناظرش با input کانولوشن وجود دارد.

```
for i, photo_size in enumerate(self.photo_sizes):
    for o in range(self.orientations):
        input_layer = Input(shape=(1, photo_size, photo_size), traces=True)
        self.network.add_layer(layer=input_layer, name="inp_%d_%d" % (o, i))

        s1 = LIFNodes(shape=(1, photo_size, photo_size), traces=True)
        self.network.add_layer(layer=s1, name="S1_%d_%d" % (o, i))
```

#### 3. C1:

۱۶ لایه دیگر که ابعاد هر لایه نصف لایه های قبلی است و بین این لایه ها و لایه های s1 MaxPoolConnection وجود داد.

```
c1 = LIFNodes(shape=(1, photo_size // 2, photo_size // 2),
               thresh=-64, traces=True)
self.network.add_layer(layer=c1, name="C1_%d_%d" % (o, i))
```

```

max_pool = MaxPool2dConnection(s1, c1, kernel_size=2, stride=2,
decay=0.0)
self.network.add_connection(max_pool, "S1_%d_%d" % (o, i), "C1_%d_%d"
% (o, i))

```

#### 4. S2:

به تعداد فیچرها ضربدر سایز عکس ها لایه وجود دارد که با کانکشن به لایه قبلی وصل اند و فیچر ها از وزن سیناپس های بین این لایه ها استخراج میشود.

```

for f in range(self.features):
    s2 = LIFNodes(shape=(1, 10, 10), traces=True)
    self.network.add_layer(layer=s2, name="S2_%d_%d" % (i, f))

    for o in range(self.orientatation):
        con = Connection(self.network.layers["C1_%d_%d" % (o, i)], s2,
update_rule=PostPre)
        self.network.add_connection(con, "C1_%d_%d" % (o, i), "S2_%d_%d" %
(i, f))
        self.network.add_monitor(Monitor(con, ["w"]), "features_%d_%d" % (i,
f))

```

#### 5. C2:

به تعداد لایه های قسمت قبل لایه با سایز یک وجود دارد که از تمام نوروں های لایه متناظرش ماکسیمم میگیرد.

```

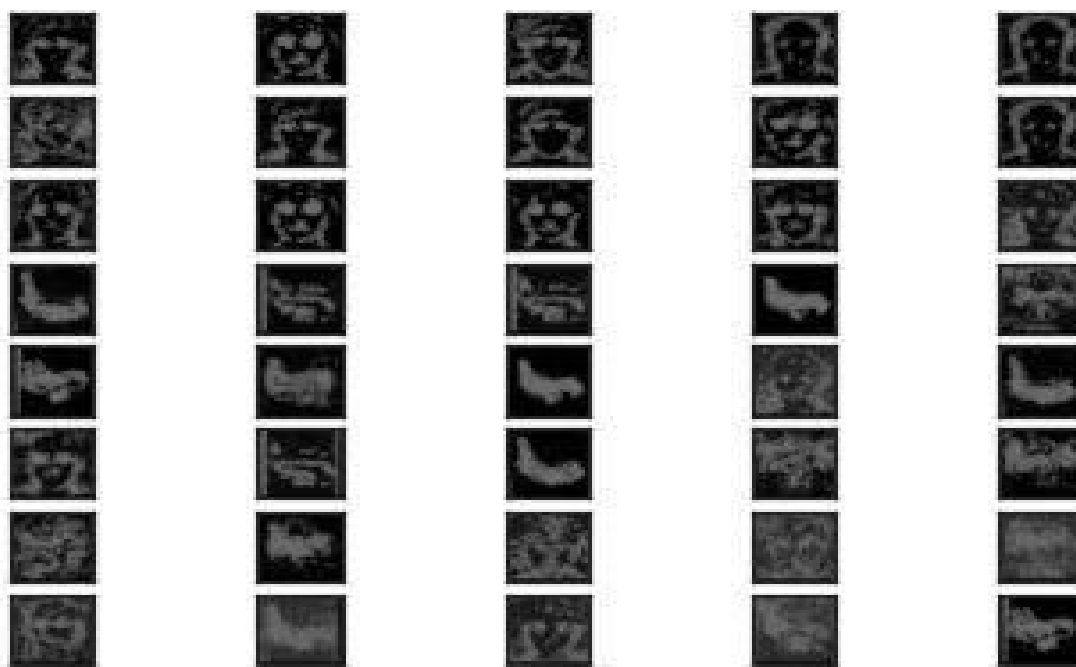
c2 = LIFNodes(shape=(1, 1, 1), thresh=-64, traces=True)
self.network.add_layer(layer=c2, name="C2_%d_%d" % (i, f))
max_pool = MaxPool2dConnection(s2, c2, kernel_size=3, decay=0)
self.network.add_connection(max_pool, "S2_%d_%d" % (i, f), "C2_%d_%d" % (i,
f))

```

و در اخر دو لایه برای تصمیم گیری اضافه شده است.

```
d1 = LIFNodes(n=self.classifier_size, traces=True)
self.network.add_layer(d1, "D1")
```

خروجی:



```
Error: 0.3672839506172839
Correctness: 0.6327160493827161
Silent Samples: 0.0
```