#ifndef CRNN\_LAYER\_H

#define CRNN\_LAYER\_H

#include "activations.h"

#include "layer.h"

#include "network.h"

#ifdef \_\_cplusplus

extern "C" {

#endif

layer make\_crnn\_layer(int batch, int h, int w, int c, int hidden\_filters, int output\_filters, int groups, int steps, int size, int stride, int dilation, int pad, ACTIVATION activation, int batch\_normalize, int xnor, int train);

void resize\_crnn\_layer(layer \*l, int w, int h);

void free\_state\_crnn(layer l);

void forward\_crnn\_layer(layer l, network\_state state);

void backward\_crnn\_layer(layer l, network\_state state);

void update\_crnn\_layer(layer l, int batch, float learning\_rate, float momentum, float decay);

#ifdef GPU

void forward\_crnn\_layer\_gpu(layer l, network\_state state);

void backward\_crnn\_layer\_gpu(layer l, network\_state state);

void update\_crnn\_layer\_gpu(layer l, int batch, float learning\_rate, float momentum, float decay, float loss\_scale);

void push\_crnn\_layer(layer l);

void pull\_crnn\_layer(layer l);

#endif

#ifdef \_\_cplusplus

}

#endif

#endif