

CLAIMS

1. A computer-implemented method for using a deep convolutional network, wherein the deep convolutional network comprises a first, a second, and a third convolutional layer, the method comprising:
 - convolving a first-layer kernel with a first first-layer stretched receptive field of a first input tensor to generate values for a first strip of a second-layer input tensor;
 - subsequently convolving a second-layer kernel with a first second-layer stretched receptive field of the second-layer input tensor to generate values for a first strip of a third-layer input tensor; and
 - subsequently convolving the first-layer kernel with a second first-layer stretched receptive field of the first input tensor to generate values for a second strip of the second-layer input tensor, wherein:
 - the first-layer kernel has a first-layer-kernel length and a first-layer-kernel width;
 - the first-layer input tensor has a first-layer-input length and a first-layer-input width;
 - the second-layer input tensor has a second-layer-input length and a second-layer-input width;
 - the second-layer kernel has a second-layer-kernel length and a second-layer-kernel width;
 - the first-layer stretched receptive fields have a length equal to the first-layer-kernel length and a first-layer-stretched-receptive-field width that is greater than the first-layer-kernel width;
 - the second-layer stretched receptive fields have a length equal to the second-layer-kernel length and a second-layer-stretched-receptive-field width that is greater than the second-layer kernel width; and
 - a stretched receptive field is a region of a tensor that has a width equal to the width of the tensor and that has a length less than the length of the tensor, and a strip of a tensor is a matrix having a length of one and a width equal to a stretched-receptive-field width.

2. The method of claim 1, further comprising:
convolving the first-layer stretched receptive fields with one or more additional first-layer kernels to generate values for corresponding additional strips of the second-layer input tensor; and
convolving the second-layer stretched receptive fields with one or more additional second-layer kernels to generate values for corresponding additional strips of the third-layer input tensor.
3. The method of claim 1, wherein:
the first-layer-stretched-receptive-field width is equal to the first-layer-input width; and
the second-layer-stretched-receptive-field width is equal to the second-layer-input width.
4. The method of claim 3, wherein:
the strips of the second-layer input tensor have a length of one and a width equal to the second-layer-input width;
the third-layer input tensor has a third-layer-input length and a third-layer-input width; and
the strips of the third-layer input tensor have a length of one and a width equal to the third-layer-input width.
5. The method of claim 1, wherein:
the deep convolutional network further comprises at least one non-convolutional layer between the first convolutional layer and the second convolutional layer; and
generating the second-layer input tensor comprises performing operations of the at least one non-convolutional layer in addition to the convolving of the first-layer input tensor.
6. The method of claim 5, wherein the at least one non-convolutional layer comprises one or more of a pooling layer, a non-linearity layer, and a normalization layer.

7. The method of claim 1, wherein:
each kernel comprises a corresponding set of weights;
the method further comprises:

reading in from a holding memory into an operational memory the weights of the first-layer kernel, the weights of the second-layer kernel; and
convolving the first-layer kernel with the first first-layer stretched receptive field comprises reading from the holding memory into the operational memory values for the first first-layer stretched receptive field without reading into the operational memory all values of the first-layer input.

8. The method of claim 7, wherein:
convolving the first-layer kernel with the first first-layer stretched receptive field is followed by releasing and reusing in the operational memory at least some of the values for the first first-layer stretched receptive field before the subsequently convolving the first-layer kernel with a second first-layer stretched receptive field of the first input tensor.

9. The method of claim 1, wherein:
the third layer is a convolutional layer;
the deep convolutional network comprises zero or more additional convolutional layers configured consecutively following the third convolutional layer for processing simultaneously with the first, second, and third convolutional layers in a set of layers; and
the set of layers comprises a cut-off layer, wherein the cut-off layer is a last layer of the set of layers, whose values get written out to a holding memory.

10. The method of claim 9, wherein the cut-off layer is dynamically determined using a greedy algorithm.

11. A computing device for using a deep convolutional network,
wherein the deep convolutional network comprises:
a first convolutional layer configured to use a first-layer kernel for convolving a first-layer input tensor to generate a second-layer input tensor;

a second convolutional layer configured to use a second-layer kernel for convolving the second-layer input tensor to generate a third-layer input tensor; and

a third layer configured to receive the third-layer input tensor; the first-layer kernel has a first-layer-kernel length and a first-layer-kernel width; and

wherein:

the first-layer input tensor has a first-layer-input length and a first-layer-input width;

the second-layer input tensor has a second-layer-input length and a second-layer-input width;

the second-layer kernel has a second-layer-kernel length and a second-layer-kernel width;

the computing device is configured to:

convolve the first-layer kernel with a first first-layer stretched receptive field of the first input tensor to generate values for a first strip of the second-layer input tensor;

subsequently convolve the second-layer kernel with a first second-layer stretched receptive field of the second input tensor to generate values for a first strip of the third-layer input tensor; and

subsequently convolve the first-layer kernel with a second first-layer stretched receptive field of the first input tensor to generate values for a second strip of the second-layer input tensor;

the first-layer stretched receptive fields have a length equal to the first-layer-kernel length and a first-layer-stretched-receptive-field width that is greater than the first-layer-kernel width;

the second-layer stretched receptive fields have a length equal to the second-layer-kernel length and a second-layer-stretched-receptive-field width that is greater than the second-layer kernel width; and

a stretched receptive field is a region of a tensor that has a width equal to the width of the tensor and that has a length less than the length of the tensor, and a strip of a tensor is a matrix having a length of one and a width equal to a stretched-receptive-field width.

12. The device of claim 11, wherein:

the first convolutional layer is configured to use one or more additional first-layer kernels in the same way as the first-layer kernel to generate values for corresponding additional strips of the second-layer input tensor; and

the second convolutional layer is configured to use one or more additional second-layer kernels in the same way as the second-layer kernel to generate values for corresponding additional strips of the third-layer input tensor.

13. The device of claim 11, wherein:

the first-layer-stretched-receptive-field width is equal to the first-layer-input width; and

the second-layer-stretched-receptive-field width is equal to the second-layer-input width.

14. The device of claim 13, wherein:

the strips of the second-layer input tensor have a length of one and a width equal to the second-layer-input width;

the third-layer input tensor has a third-layer-input length and a third-layer-input width; and

the strips of the third-layer input tensor have a length of one and a width equal to the third-layer-input width.

15. The device of claim 11, wherein:

the deep convolutional network further comprises at least one non-convolutional layer between the first convolutional layer and the second convolutional layer; and

generating the second-layer input tensor comprises performing operations of the at least one non-convolutional layer in addition to the convolving of the first-layer input tensor.

16. The device of claim 15, wherein the at least one non-convolutional

layer comprises one or more of a pooling layer, a non-linearity layer, and a normalization layer.

17. The device of claim 11, wherein:
each kernel comprises a corresponding set of weights;
the device is further configured to read in from a holding memory into an operational memory the weights of the first-layer kernel, the weights of the second-layer kernel; and
in convolving the first-layer kernel with the first first-layer stretched receptive field, the device is configured to read from the holding memory into the operational memory values for the first first-layer stretched receptive field without reading into the operational memory all values of the first-layer input.

18. The device of claim 17, wherein the device is configured to, following convolution of the first-layer kernel with the first first-layer stretched receptive field, release and reuse in the operational memory at least some of the values for the first first-layer stretched receptive field before subsequently convolving the first-layer kernel with a second first-layer stretched receptive field of the first input tensor.

19. The device of claim 11, wherein:
the third layer is a convolutional layer;
the deep convolutional network comprises zero or more additional convolutional layers configured consecutively following the third convolutional layer for processing simultaneously with the first, second, and third convolutional layers in a set of layers;
the set of layers comprises a cut-off layer wherein the cut-off layer is a last layer of the set of layers, whose values get written out to a holding memory.

20. The device of claim 19, wherein the device is configured to dynamically determining the cut-off layer using a greedy algorithm.

21. A non-transitory computer-readable medium comprising instructions that, when executed by at least one processor, cause the processor to perform the method of any one of claims 1 to 9.