

Optical Character Recognition (OCR)

- **CV**

Tesseract 3 (A*)

Binarizuje, naporcuje obrazok na kúsky a snaží sa z nich poskladať symboly

- **CNN+RNN**

Pre každý pixel povie pravdepodobnosť že je súčasťou textu, z oblastí z vysokou pravdepodobnosťou spraví obdĺžniky a tieto rozpoznáva rekurentnou neurónovou sieťou

- **Holistic**

Základuje obraz na príznakovú mapu a z nej generuje texty (transformer typu encoder-decoder)

Vstup

technical details are too complex to cover in the book itself.

In teaching our courses, we have found it useful for the students to attempt a number of small implementation projects, which often build on one another, in order to get them used to working with real-world images and the challenges that these present. The students are then asked to choose an individual topic for each of their small-group, final projects. (Sometimes these projects even turn into conference papers!) The exercises at the end of each chapter contain numerous suggestions for smaller mid-term projects, as well as more open-ended problems whose solutions are still active research topics. Wherever possible, I encourage students to try their algorithms on their own personal photographs, since this better motivates them, often leads to creative variants on the problems, and better acquaints them with the variety and complexity of real-world imagery.

In formulating and solving computer vision problems, I have often found it useful to draw inspiration from three high-level approaches:

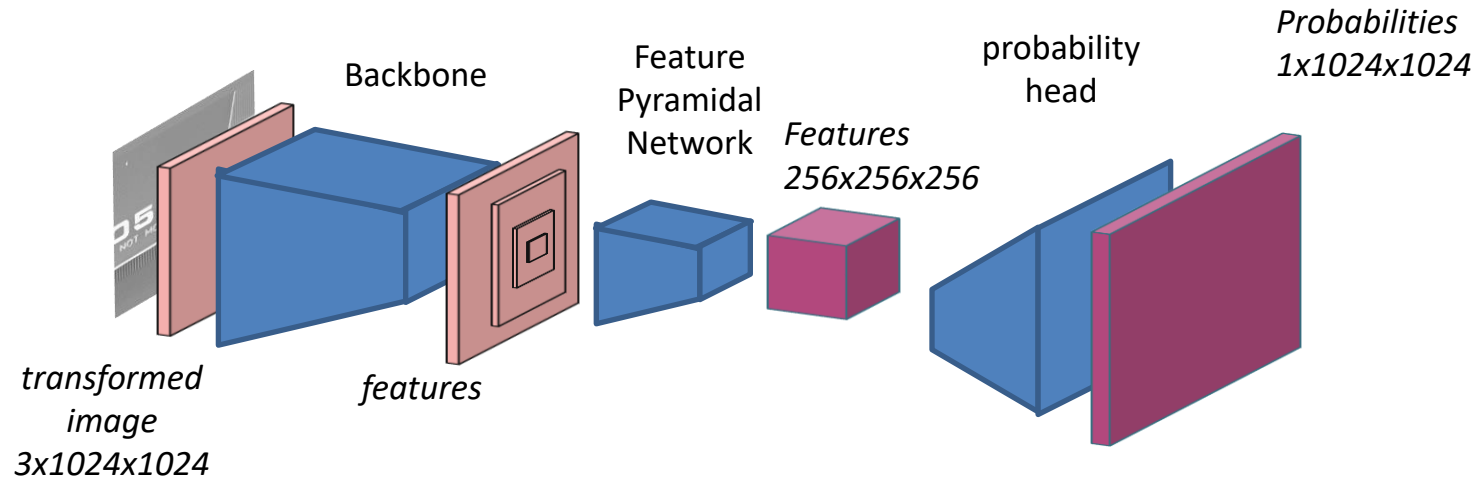
CNN+RNN

OCR tohto druhu sa skladá z dvoch hlbokých modelov:

- Detektor textu (konvolučná sieť)
- Rozpoznávač textu (dvojsmerná rekurentná sieť)

Výstup prvého je so vstupom druhého prepojený
algoritmami klasického počítačového videnia

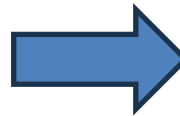
Detektor textu



technical details are too complex to cover in the book itself.

In teaching our courses, we have found it useful for the students to attempt a number of small implementation projects, which often build on one another, in order to get them used to working with real-world images and the challenges that these present. The students are then asked to choose an individual topic for each of their small-group, final projects. (Sometimes these projects even turn into conference papers!) The exercises at the end of each chapter contain numerous suggestions for smaller mid-term projects, as well as more open-ended problems whose solutions are still active research topics. Wherever possible, I encourage students to try their algorithms on their own personal photographs, since this better motivates them, often leads to creative variants on the problems, and better acquaints them with the variety and complexity of real-world imagery.

In formulating and solving computer vision problems, I have often found it useful to draw inspiration from three high-level approaches:



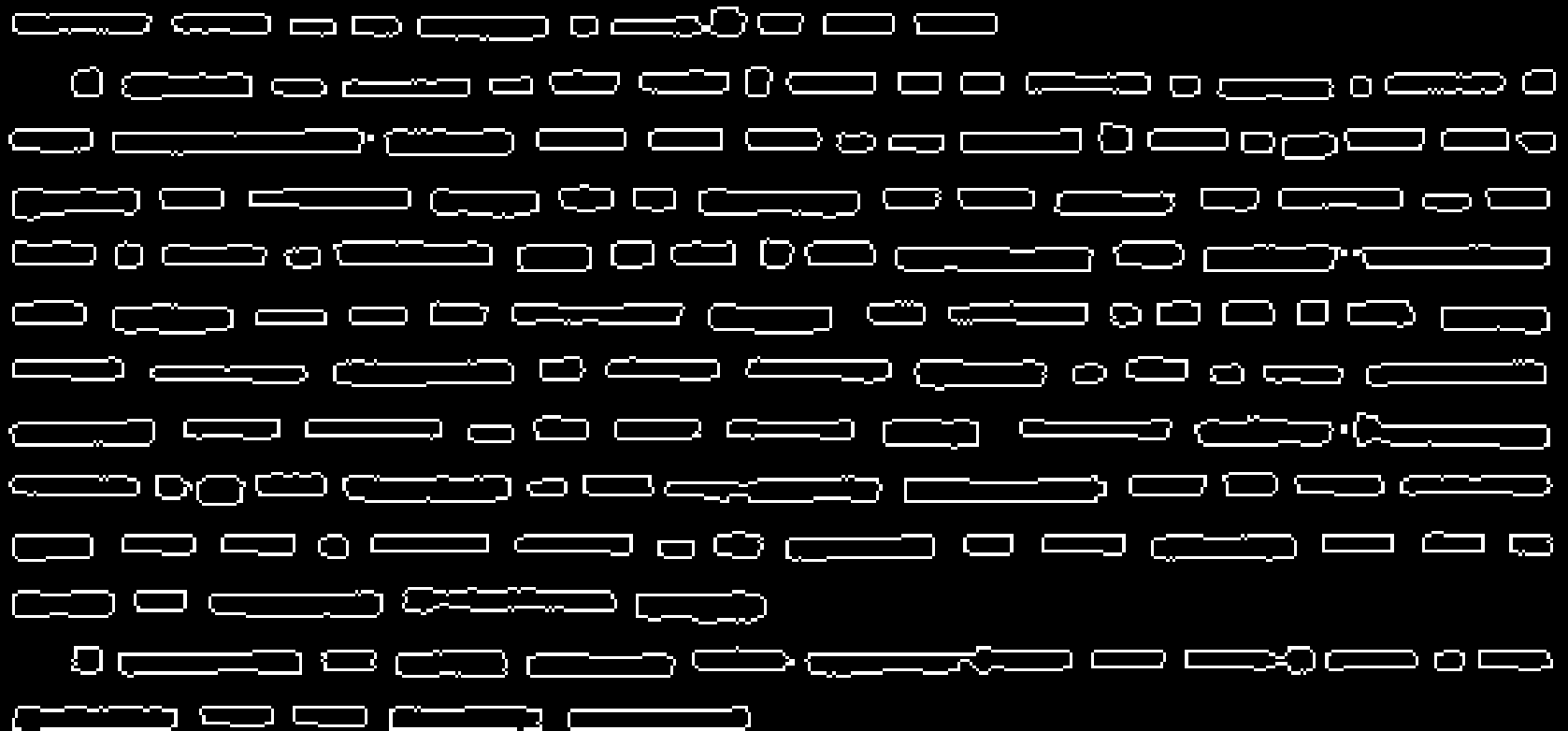
Výstup detektora

1. **Introduction**
 2. **Background**
 3. **Methodology**
 4. **Results**
 5. **Discussion**
 6. **Conclusion**
 7. **References**
 8. **Appendix**
 9. **Figure 1**
 10. **Figure 2**
 11. **Figure 3**
 12. **Figure 4**
 13. **Figure 5**
 14. **Figure 6**
 15. **Figure 7**
 16. **Figure 8**
 17. **Figure 9**
 18. **Figure 10**
 19. **Figure 11**
 20. **Figure 12**
 21. **Figure 13**
 22. **Figure 14**
 23. **Figure 15**
 24. **Figure 16**
 25. **Figure 17**
 26. **Figure 18**
 27. **Figure 19**
 28. **Figure 20**
 29. **Figure 21**
 30. **Figure 22**
 31. **Figure 23**
 32. **Figure 24**
 33. **Figure 25**
 34. **Figure 26**
 35. **Figure 27**
 36. **Figure 28**
 37. **Figure 29**
 38. **Figure 30**
 39. **Figure 31**
 40. **Figure 32**
 41. **Figure 33**
 42. **Figure 34**
 43. **Figure 35**
 44. **Figure 36**
 45. **Figure 37**
 46. **Figure 38**
 47. **Figure 39**
 48. **Figure 40**
 49. **Figure 41**
 50. **Figure 42**
 51. **Figure 43**
 52. **Figure 44**
 53. **Figure 45**
 54. **Figure 46**
 55. **Figure 47**
 56. **Figure 48**
 57. **Figure 49**
 58. **Figure 50**
 59. **Figure 51**
 60. **Figure 52**
 61. **Figure 53**
 62. **Figure 54**
 63. **Figure 55**
 64. **Figure 56**
 65. **Figure 57**
 66. **Figure 58**
 67. **Figure 59**
 68. **Figure 60**
 69. **Figure 61**
 70. **Figure 62**
 71. **Figure 63**
 72. **Figure 64**
 73. **Figure 65**
 74. **Figure 66**
 75. **Figure 67**
 76. **Figure 68**
 77. **Figure 69**
 78. **Figure 70**
 79. **Figure 71**
 80. **Figure 72**
 81. **Figure 73**
 82. **Figure 74**
 83. **Figure 75**
 84. **Figure 76**
 85. **Figure 77**
 86. **Figure 78**
 87. **Figure 79**
 88. **Figure 80**
 89. **Figure 81**
 90. **Figure 82**
 91. **Figure 83**
 92. **Figure 84**
 93. **Figure 85**
 94. **Figure 86**
 95. **Figure 87**
 96. **Figure 88**
 97. **Figure 89**
 98. **Figure 90**
 99. **Figure 91**
 100. **Figure 92**
 101. **Figure 93**
 102. **Figure 94**
 103. **Figure 95**
 104. **Figure 96**
 105. **Figure 97**
 106. **Figure 98**
 107. **Figure 99**
 108. **Figure 100**
 109. **Figure 101**
 110. **Figure 102**
 111. **Figure 103**
 112. **Figure 104**
 113. **Figure 105**
 114. **Figure 106**
 115. **Figure 107**
 116. **Figure 108**
 117. **Figure 109**
 118. **Figure 110**
 119. **Figure 111**
 120. **Figure 112**
 121. **Figure 113**
 122. **Figure 114**
 123. **Figure 115**
 124. **Figure 116**
 125. **Figure 117**
 126. **Figure 118**
 127. **Figure 119**
 128. **Figure 120**
 129. **Figure 121**
 130. **Figure 122**
 131. **Figure 123**
 132. **Figure 124**
 133. **Figure 125**
 134. **Figure 126**
 135. **Figure 127**
 136. **Figure 128**
 137. **Figure 129**
 138. **Figure 130**
 139. **Figure 131**
 140. **Figure 132**
 141. **Figure 133**
 142. **Figure 134**
 143. **Figure 135**
 144. **Figure 136**
 145. **Figure 137**
 146. **Figure 138**
 147. **Figure 139**
 148. **Figure 140**
 149. **Figure 141**
 150. **Figure 142**
 151. **Figure 143**
 152. **Figure 144**
 153. **Figure 145**
 154. **Figure 146**
 155. **Figure 147**
 156. **Figure 148**
 157. **Figure 149**
 158. **Figure 150**
 159. **Figure 151**
 160. **Figure 152**
 161. **Figure 153**
 162. **Figure 154**
 163. **Figure 155**
 164. **Figure 156**
 165. **Figure 157**
 166. **Figure 158**
 167. **Figure 159**
 168. **Figure 160**
 169. **Figure 161**
 170. **Figure 162**
 171. **Figure 163**
 172. **Figure 164**
 173. **Figure 165**
 174. **Figure 166**
 175. **Figure 167**
 176. **Figure 168**
 177. **Figure 169**
 178. **Figure 170**
 179. **Figure 171**
 180. **Figure 172**
 181. **Figure 173**
 182. **Figure 174**
 183. **Figure 175**
 184. **Figure 176**
 185. **Figure 177**
 186. **Figure 178**
 187. **Figure 179**
 188. **Figure 180**
 189. **Figure 181**
 190. **Figure 182**
 191. **Figure 183**
 192. **Figure 184**
 193. **Figure 185**
 194. **Figure 186**
 195. **Figure 187**
 196. **Figure 188**
 197. **Figure 189**
 198. **Figure 190**
 199. **Figure 191**
 200. **Figure 192**
 201. **Figure 193**
 202. **Figure 194**
 203. **Figure 195**
 204. **Figure 196**
 205. **Figure 197**
 206. **Figure 198**
 207. **Figure 199**
 208. **Figure 200**
 209. **Figure 201**
 210. **Figure 202**
 211. **Figure 203**
 212. **Figure 204**
 213. **Figure 205**
 214. **Figure 206**
 215. **Figure 207**
 216. **Figure 208**
 217. **Figure 209**

It appears the Bureau has not failed to take the necessary steps to ensure a complete and accurate investigation. Indeed, the fact that the Bureau has not failed to take the necessary steps to ensure a complete and accurate investigation is a testament to the Bureau's commitment to the highest standards of law enforcement. The Bureau's commitment to the highest standards of law enforcement is a testament to the Bureau's commitment to the highest standards of law enforcement.

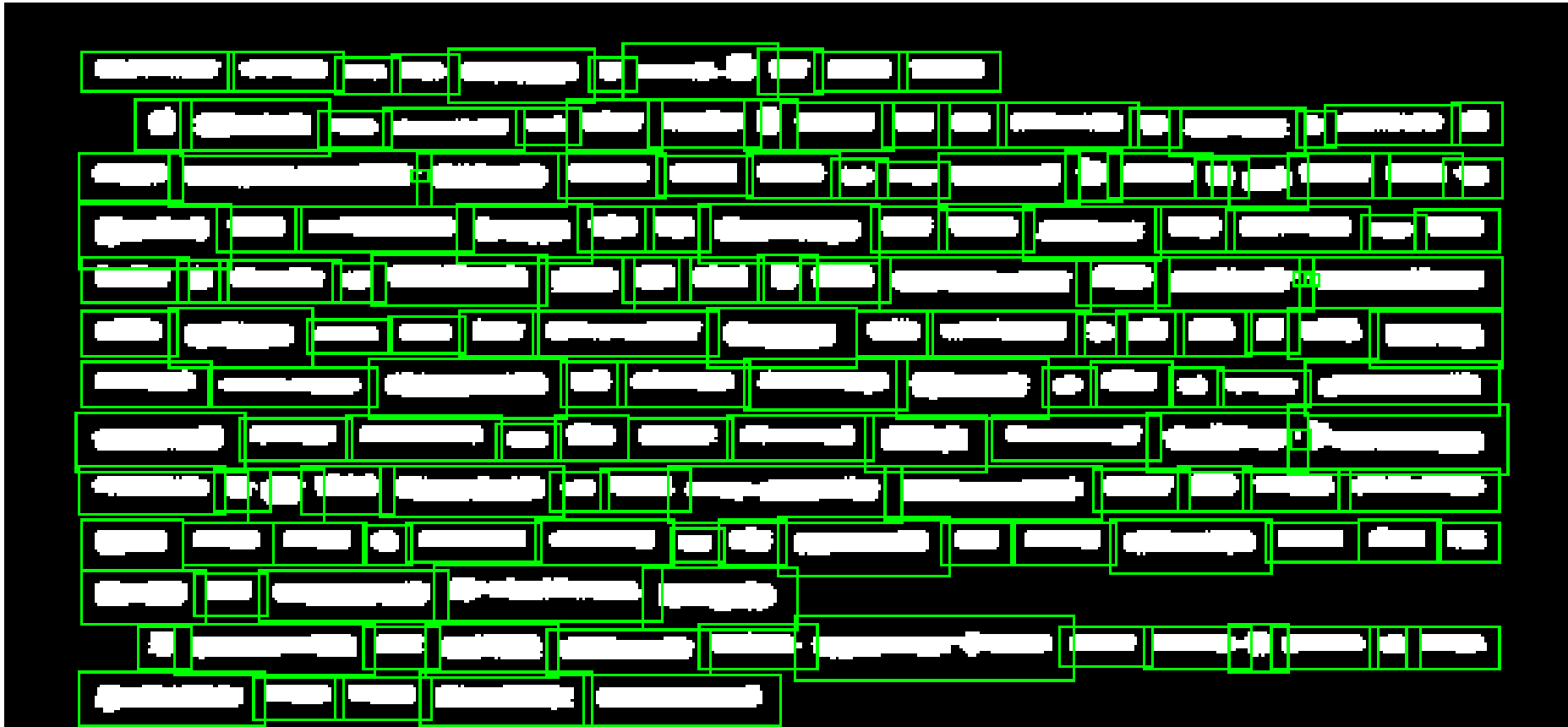
1. **Identify the main components of the system.** The system consists of a **client** and a **server**. The client is responsible for sending requests to the server, and the server is responsible for processing these requests and returning responses.

Kontúry



Kontúry môžeme aproximovať štyrmi alebo viacerými bodmi

Obdĺžniky alebo Polygóny



Pritom oblasti zodpovedajúco zväčšíme, aby sme zachytili celý text

Výsledok spracovania detekcie

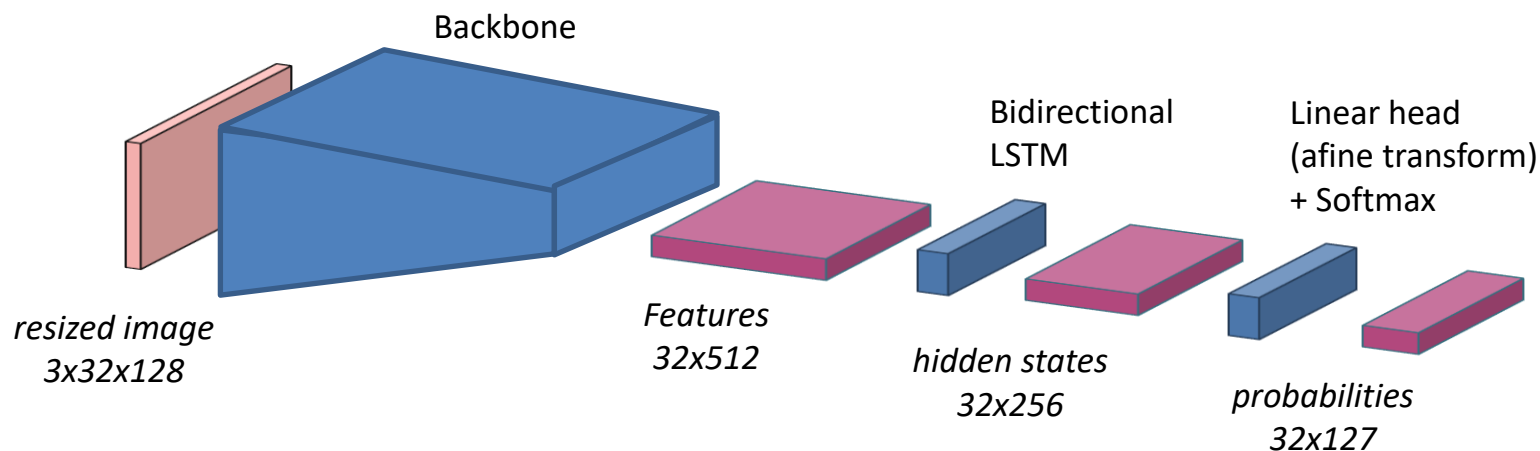
technical details are too complex to cover in the book itself.

In teaching our courses, we have found it useful for the students to attempt a number of small implementation projects, which often build on one another, in order to get them used to working with real-world images and the challenges that these present. The students are then asked to choose an individual topic for each of their small-group, final projects. (Sometimes these projects even turn into conference papers!) The exercises at the end of each chapter contain numerous suggestions for smaller mid-term projects, as well as more open-ended problems whose solutions are still active research topics. Wherever possible I encourage students to try their algorithms on their own personal photographs, since this better motivates them, often leads to creative variants on the problems, and better acquaints them with the variety and complexity of real-world imagery.

In formulating and solving computer vision problems, I have often found it useful to draw inspiration from three high-level approaches:

Každý obĺžnik či polygón vyjmene z textu a budeme rozpoznávať

Rozpoznávač



vocab (126 characters):

0123456789abcdefghijklmnopqrstuvwxyzABCD
EFGHIJKLMNOPQRSTUVWXYZ!"#\$%&'()*+,-
. / : ; < = > ? @ [\] ^ _ ` { | } ~ ° £ € ¥ ¢ ₪
à â é ê ë ï î ò ù û ü ç À Â É Ê Ë Ì Î Ï Ô Ù Ú Ü Ç

Rozpoznávanie

- Vyrežeme kúsok textu

approaches:

- Upravíme jeho veľkosť na 32x128

approaches:

- Tento obraz premeníme na mapu príznačkov 32x512
- Tú dekodujeme cez LSTM (typ RNN) na skryté stavy 32x256
- Lineárnou hlavou ich premeníme na 32xN logitov, kde N je počet znakov + 1 (BLANK), aplikujeme softmax
- Cez CTC z pravdepodobností vyberieme symboly

Problém s rýchlosťou

- Dvojsmerná (bidirectional) RNN je schopná pracovať na texte rôznej dĺžky (je natáhovacia)
- Ale nevie pracovať na textoch rôznej dĺžky naraz v jednej dávke
- Preto DocTR radšej naseká texty tak, aby mali menej ako 32 symbolov a používame rovnako veľký vstup $3 \times 32 \times 128$
- (Ale Pytorch vie pustiť rôzne dĺžky naraz, akurát vyžaduje aby bola dávka utriedená podľa dĺžky zostupne takže treba urobiť správnu permutáciu)

Výstup

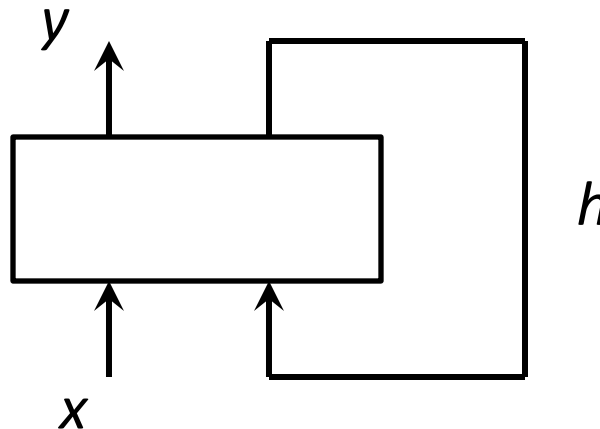
technical details are too complex to cover in the book itself.

In teaching our courses, we have found it useful for the students to attempt a number of small implementation projects, which often build on one another, in order to get them used to working with real-world images and the challenges that these present. The students are then asked to choose an individual topic for each of their small-group, final projects. (Sometimes these projects even turn into conference papers!) The exercises at the end of each chapter contain numerous suggestions for smaller mid-term projects, as well as more open-ended problems whose solutions are still active research topics. Wherever possible, I encourage students to try their algorithms on their own personal photographs, since this better motivates them, often leads to creative variants on the problems, and better acquaints them with the variety and complexity of real-world imagery.

In formulating and solving computer vision problems, I have often found it useful to draw inspiration from three high-level approaches:

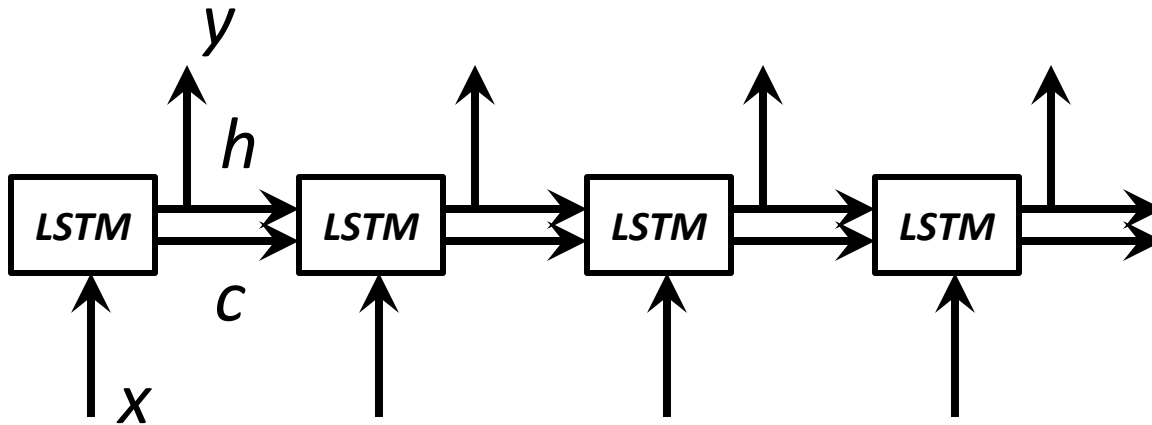
Rekurentná neurónová sieť

- Sieť inšpirovaná spracúvaním postupnosti dát v čase
- V každom okamihu dostáva nový vstup, dáva nový výstup a skrytý stav sa zapamätá a bude pridaný na vstup v ďalšom okamihu



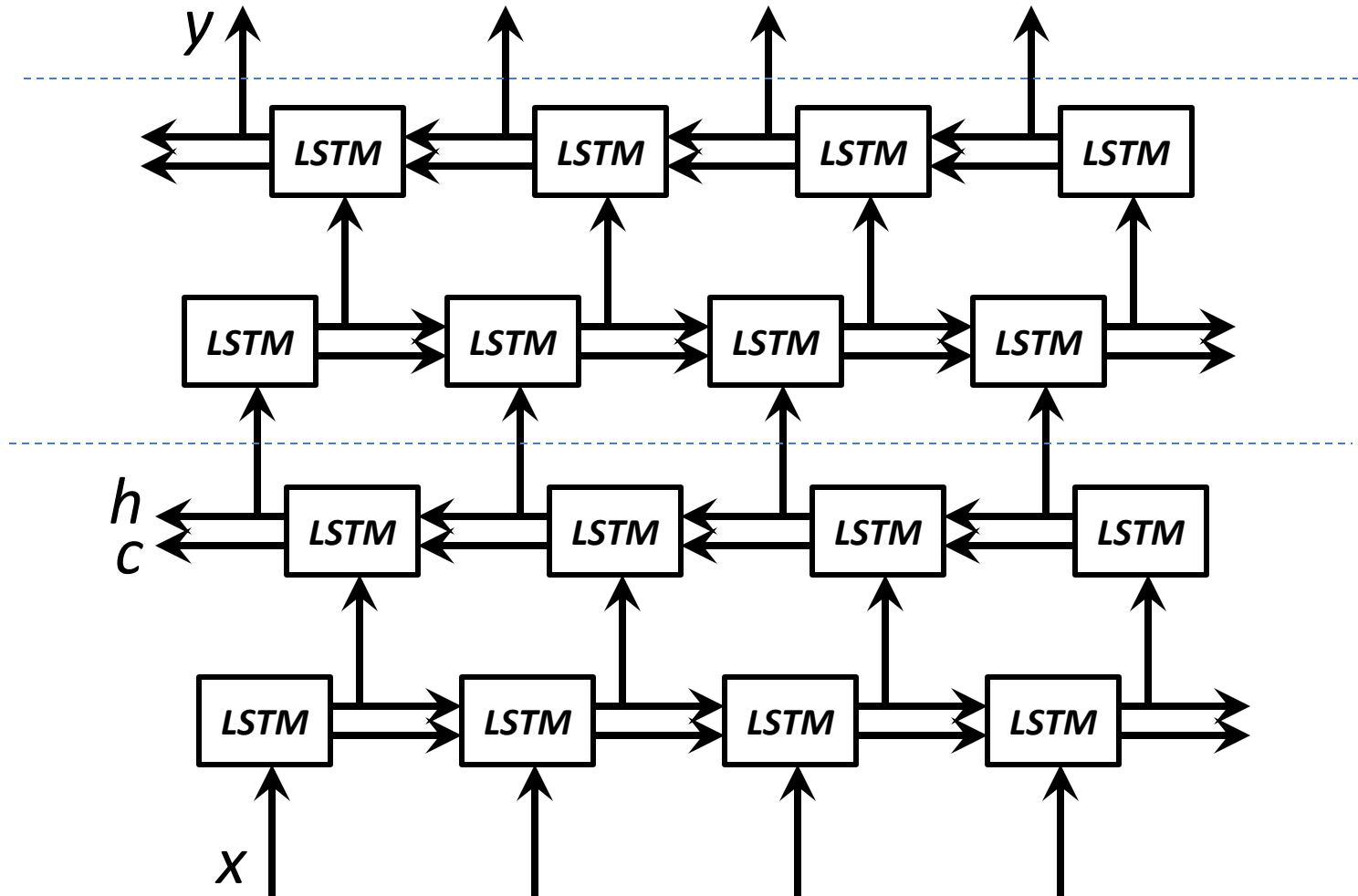
RNN

- Reálne sa nepúšťa na dáta postupne v čase, ale na konci naraz



- počet LSTM modulov je daný dĺžkou vstupu

Bidirectional RNN

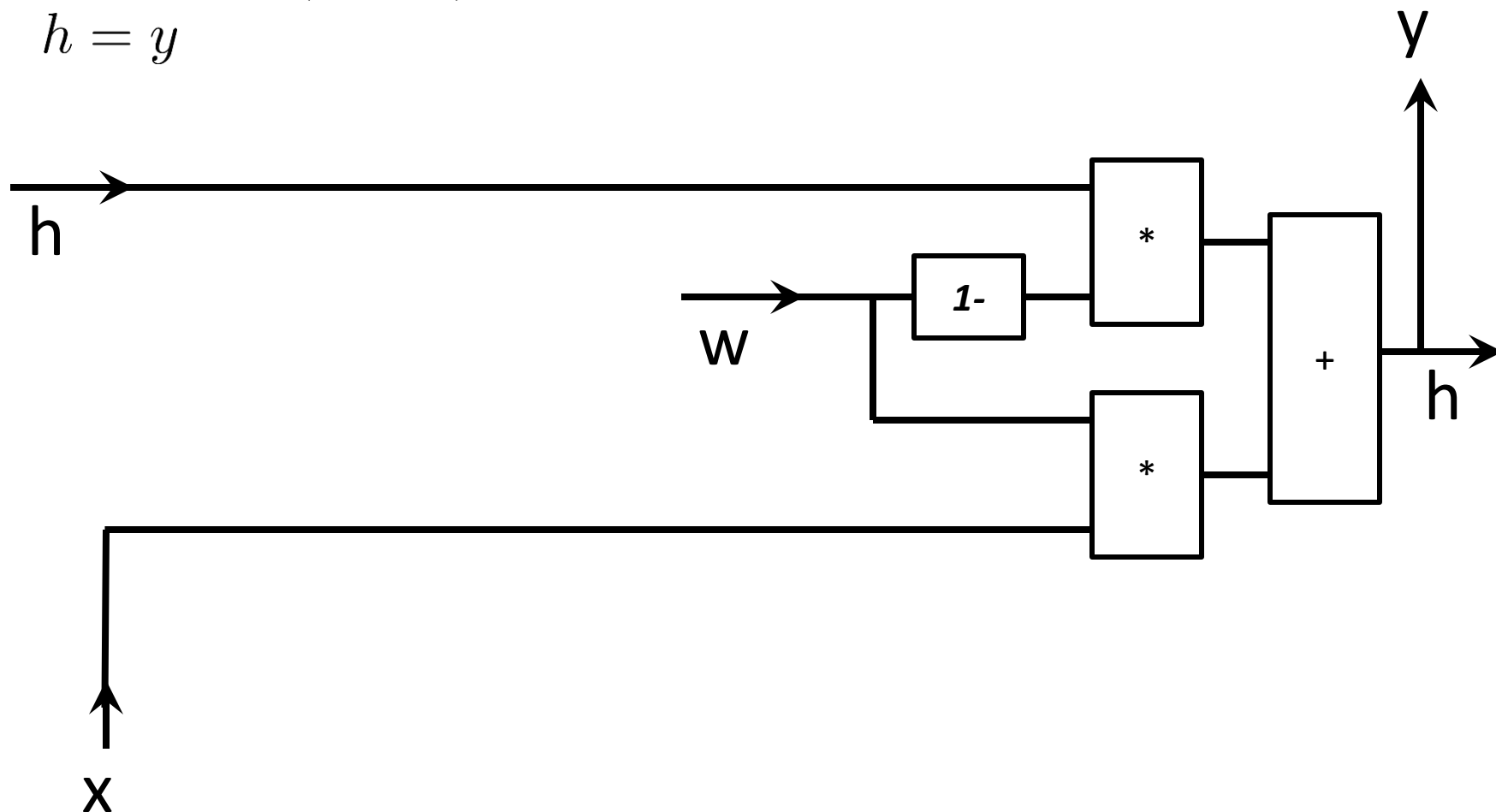


x: -1, -1, -1, -1, -1, -1, 1, 1, 1, 1, -1, 1, 1, 1, 1, 1, 1, -1, 1, 1, 1, 1, 1...

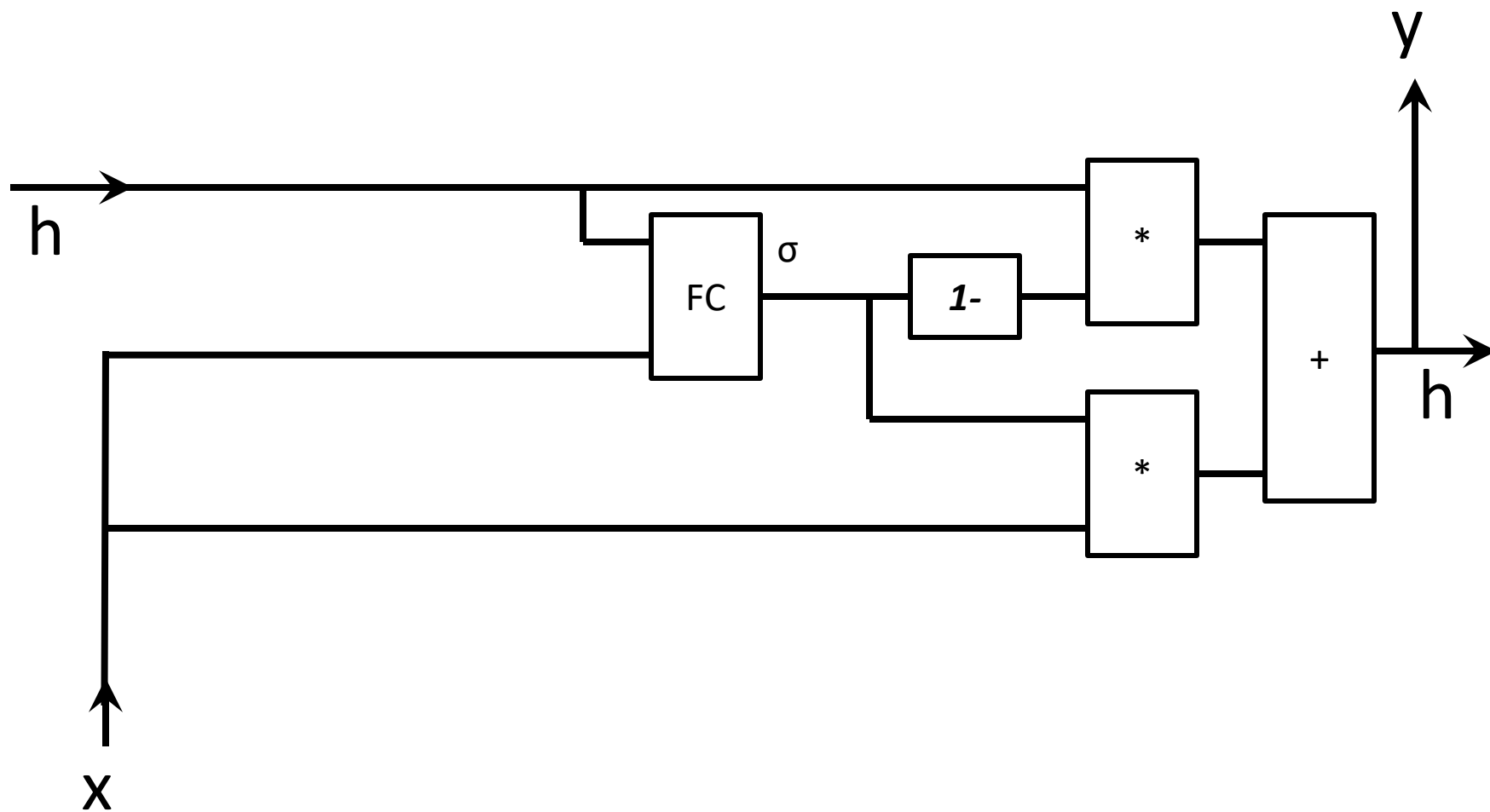
y: 0, -0.4, -0.7, -0.9, -1, -0.9, -0.7, -0.4, 0.1, 0.5, 0.9, 1, 1, 1, 1, ...

$$y = wx + (1 - w)h$$

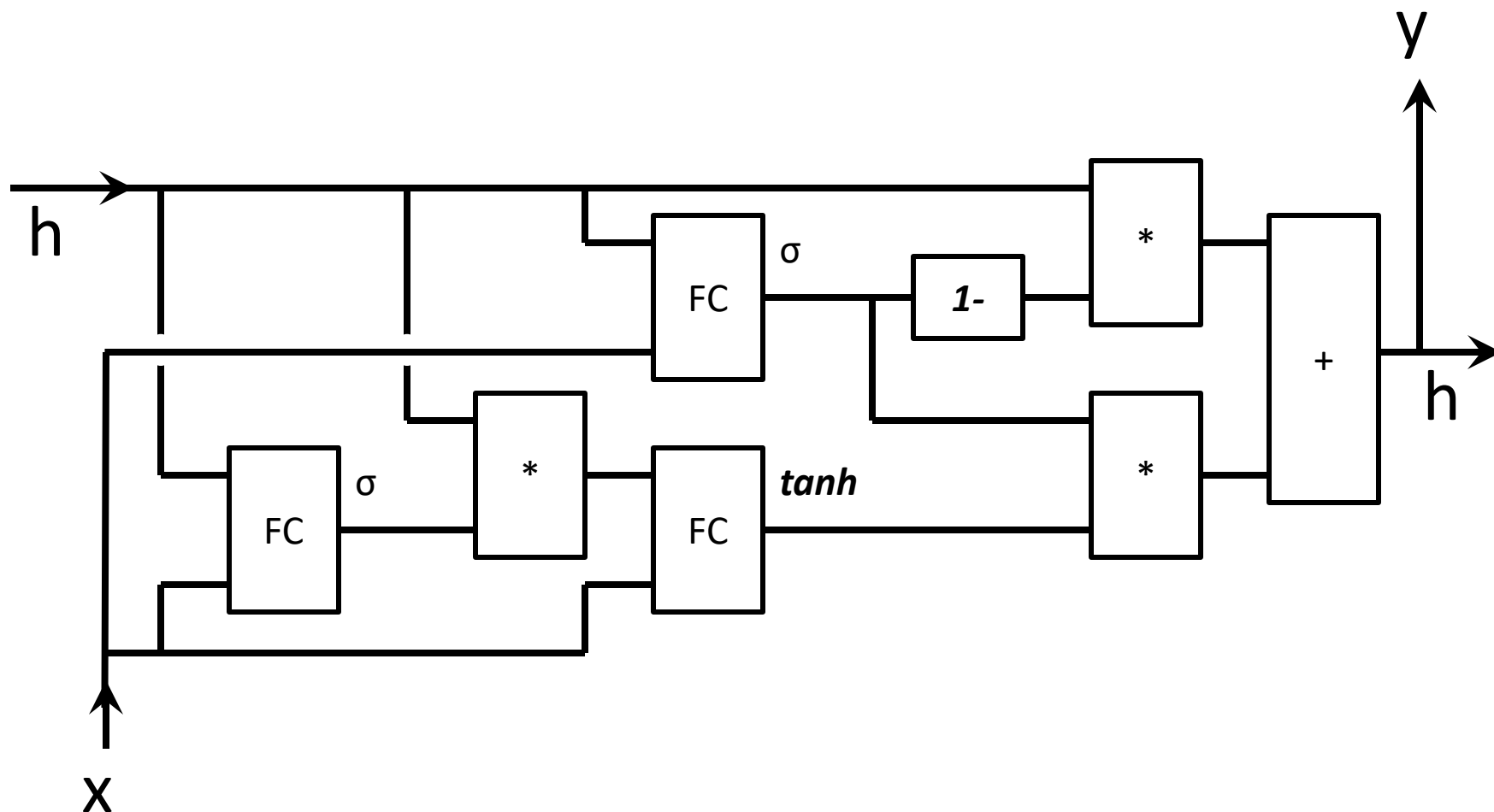
$$h = y$$



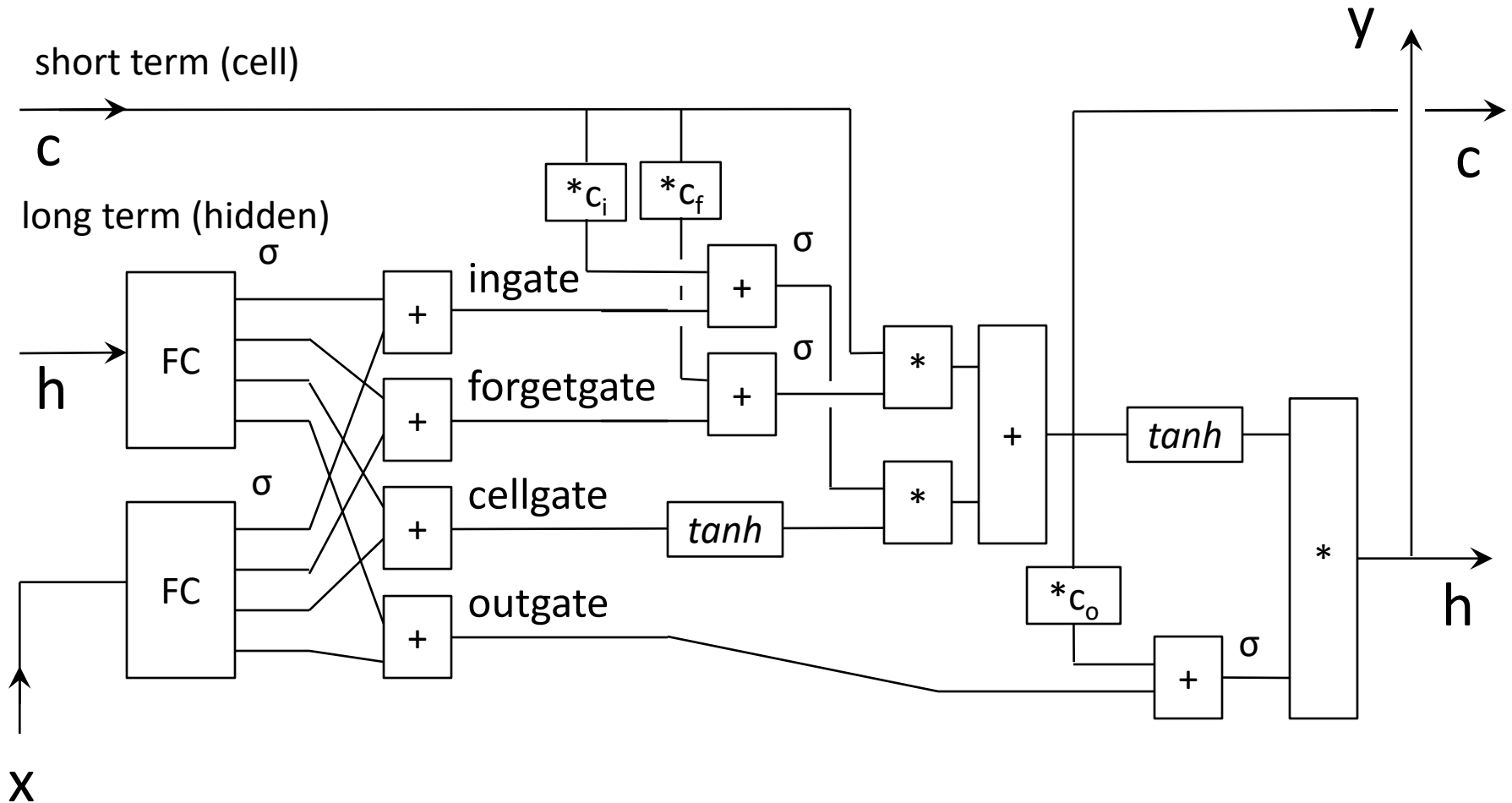
dimension of x, y and h can be > 1



Gated recurrent unit (GRU)



Long Short-Term Memory (LSTM)



Connectionist Temporal Classification (CTC)

- Chybová funkcia
- Počíta sumu pravdepodobností pre všetky možné zarovnaní
- Zarovnanie je cesta cez tenzor pravdepodobností, ktorá pod odstránení BLANKu dáva požadovanú sekvenciu; pravdepodobnosť cesty je súčinom pravdepodobností krokov
- Počíta sa efektívne cez dynamické programovanie

Connectionist Temporal Classification (CTC)

- Ako vedľajší produkt vie vypočítať najpravdepodobnejšiu cestu, takže ju je možné použiť aj na dekódovanie pri inferencii (beam search)
- Po natrénovaní však celkom dobre funguje aj greedy: pre každý výstup sa vezme najpravdepodobnejší symbol
- Rozdiel: beam nájde cestu s najväčším súčinom pravdepodobností, greedy s najväčším súčtom