

UDC 004.9

SCOPUS CODE 1705

<https://doi.org/10.36073/1512-0996-2020-3-33-56>

კონვოლუციური ნეირონული ქსელები

არჩილ ფრანგიშვილი	კომპიუტერული ინჟინერიის დეპარტამენტი, საქართველოს უნივერსიტეტი, საქართველო, 0160, თბილისი, მ. კოსტავას 77 E-mail: a_prangi@gtu.ge	ტექნიკური
ოლეგ ნამიჩეიშვილი	კომპიუტერული ინჟინერიის დეპარტამენტი, საქართველოს უნივერსიტეტი, საქართველო, 0160, თბილისი, მ. კოსტავას 77 E-mail: o.namicheishvili@gtu.ge	ტექნიკური
მიხეილ რამაზაშვილი	კომპიუტერული ინჟინერიის დეპარტამენტი, საქართველოს უნივერსიტეტი, საქართველო, 0160, თბილისი, მ. კოსტავას 77 E-mail: m.ramazashvili@gtu.ge	ტექნიკური

რეცენზენტები:

ზ. გასიტაშვილი, სტუ-ის ინფორმატიკისა და მართვის სისტემების ფაკულტეტის პროფესორი
E-mail: zur_gas@gtu.ge

ნ. მჭედლიშვილი, სტუ-ის ინფორმატიკისა და მართვის სისტემების ფაკულტეტის პროფესორი
E-mail: galoba47@mail.ru

ანოტაცია. ღრმა სწავლების ასპექტში სტატია განიხილავს ე.წ. კონვოლუციურ (ხვეული) ნეირონულ ქსელებს. სახელდობრ, ნაშრომში სისტემატური ფორმით გადმოცემულია შემდეგი საკითხები: კონვოლუციის («ნახვევის») ოპერაცია, ხვეული შრის ზოგადი სტრუქტურა, ხვეული ქსელის შემავალი და გამომავალი მონაცემები, შეცდომის უკუგავრცელების მეთოდი ხვეული ქსელებისათვის, დასასწავლი პარამეტრების რაოდენობის განსაზღვრა, ხვეული ქსელების შესანახად აუცილებელი მეხსიერების მოცულობის შეფასება, ხვეული ნეირონული ქსელების ხელნაწერ ციფრთა კლასიფიკაციის ამოცანაში გამო-

ყენების მაგალითი, ხვეული ქსელების აგების პრინციპები, მოდელის დეგრადაციის პრობლემა, ღრმა (სიღრმისეული) ნარჩენი ქსელები და ხელნაწერ ციფრთა კლასიფიკაციის ამოცანაში ღრმა ნარჩენი ქსელების გამოყენების მაგალითი. საგანგებოდ მახვილდება ყურადღება ხვეული ნეირონული ქსელების არქიტექტურათა განვითარებაზე.

საკვანძო სიტყვები: ნახვევის ოპერაცია, შეცდომის უკუგავრცელების მეთოდი, დასასწავლი პარამეტრები, ხელნაწერი ციფრების კლასიფიკაცია, მოდელის დეგრადაცია, ღრმა (სიღრმისეული) ნარჩენი ქსელი.

შესავალი

ამჟამად ხელოვნური ნეირონული ქსელი ფართოდ გამოიყენება სხვადასხვა ამოცანის გადასაჭრელად რეალურ ცხოვრებაში. ისეთი პრობლემები, როგორცაა მეტყველების გამოცნობა ან სიმბოლოების ოპტიკური გამოცნობა, დღეს წყდება დიდი სიზუსტით ნეირონული ქსელების გამოყენების საფუძველზე. მოცემულ ნაშრომში ნეირონული ქსელის ერთ-ერთი სპეციალური არქიტექტურა განიხილება – ხვეული (კონვოლუციური) ნეირონული ქსელი, ასევე მისი სტრუქტურა და გამოყენება მონაცემთა სხვადასხვა ტიპის კლასიფიკაციისათვის.

ძირითადი ნაწილი

კონვოლუციის ოპერაცია და კონვოლუციური ქსელის ცნება

კონვოლუციური ნეირონული ქსელები ისეთი ნეირონული ქსელების სახეა, რომლებიც თუნდაც ერთ-ერთ თავიანთ შრეებზე გამოიყენებს გარდასახვად კონვოლუციის («ნახვევის») ოპერაციას. კონვოლუცია არის ორ ნამდვილიწმნა ფუნქციაზე განხორციელებული ოპერაცია. ამ ოპერაციის მუშაობის გასაგებად მივმართოთ [3] ნაშრომში განხილულ მაგალითს.

დავუშვათ, რომ მიმდინარეობს კოსმოსურ საფრენ აპარატზე დაკვირვება ლაზერის მეშვეობით. ლაზერი უზრუნველყოფს $x(t)$ გამომავალ სიგნალს - ხომალდის მდგომარეობას დროის t მომენტში. ხმაურით ნაკლებად დაბინძურებული სიგნალის მისაღებად და ხომალდის მდებარეობის უფრო ზუსტად შესაფასებლად აუცილებელია რამდენიმე ახლო გაზომვის გასაშუალება. ამასთან, რაც უფრო ადრე ან რაც უფრო გვიან მიმდინარე მომენტის მი-

მართ განხორციელებულია გაზომვა, მით უფრო ნაკლებ გავლენას უნდა ახდენდეს მოცემული გაზომვა დასადგენ საშუალოზე. ამიტომ აუცილებელია გამოითვალოს შეწონილი საშუალო

$$s(t) = \int x(a)w(t-a)da.$$

სწორედ რომ ამ ოპერაციას ეწოდება კონვოლუცია და იგი შემდეგნაირად აღინიშნება:

$$s(t) = \langle x * w \rangle (t).$$

კონვოლუციური ნეირონული ქსელების ტერმინოლოგიაში კონვოლუციის პირველ ელემენტს - $x(\cdot)$ ფუნქციას – ეწოდება *შესასვლელი*, ხოლო მეორე არგუმენტს - $w(\cdot)$ ფუნქციას – *კონვოლუციის ბირთვი*. ამასთან ერთად გამოსასვლელს ეწოდება *ნიშნების რუკა* (feature map). მონაცემები კომპიუტერში დისკრეტულია და გაზომვები ხდება დროის გარკვეული ინტერვალით (ვთქვათ, ერთჯერ წამში), ამიტომ, როგორც წესი, განიხილება დისკრეტული კონვოლუცია

$$s(t) = \langle x * w \rangle (t) = \sum_{a=-\infty}^{\infty} x(a)w(t-a).$$

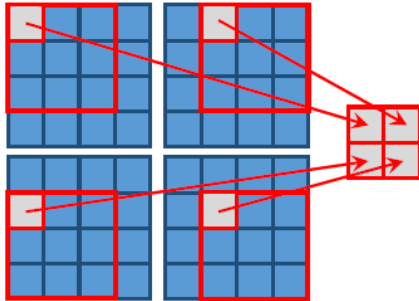
მანქანური სწავლების ამოცანებში შესასვლელი არის მონაცემთა მრავალგანზომილებიანი მასივი (ტენზორი), ხოლო ბირთვი – ისეთი პარამეტრების მასივია, რომლებიც შეირჩევა სწავლებით. მაგალითად, თუ შესასვლელზე არის ორგანზომილებიანი I გამოსახულება და K ბირთვი, მაშინ კონვოლუციის ოპერაცია შემდეგნაირად გამოიყურება:

$$s(i, j) = \langle I * K \rangle (i, j) = \sum_m \sum_n I(m, n)K(i-m, j-n).$$

კონვოლუციის ფუნქცია არის კომპუტაციური, ასოციაციური და წრფივი. ღრმა (სიდრმისეული) სწავლების ბიბლიოთეკათა უმრავლესობა ახდენს

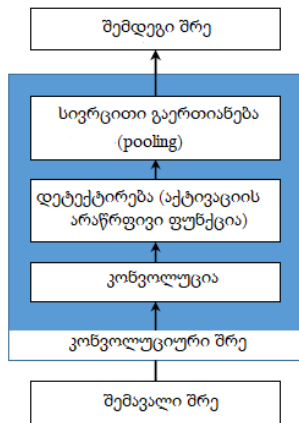
კროს-კორელაციის ბმული ფუნქციის რეალიზებას და სწორედ მას უწოდებენ კონვოლუციას (სურ. 1):

$$s(i, j) = \langle I * K \rangle(i, j) = \sum_m \sum_n I(i+m, j+n) K(m, n).$$



სურ. 1. კონვოლუციის ოპერაციის მუშაობის სქემა 4×4 პიქსელის გარჩევითობის გამოსახულებაზე 3×3 ზომის ბირთვით (წითელი ჩარჩო შეესაბამება ფილტრის მდგომარეობას შემოვლისას; ნაცრისფერი კვადრატი წითელ ჩარჩოში - ფილტრის წამყვან პოზიციას)

უნდა აღინიშნოს, რომ შემავალი ტენზორის საზღვრების დამუშავება შეიძლება სხვადასხვანაირად ხდებოდეს. როგორც წესი, ტენზორის შევსება ხდება ნულოვანი საზღვრებით. მათ საჭიროებას განსაზღვრავს კონვოლუციის გამოთვლის აუცილებლობა სასაზღვრო წერტილებზე.



სურ. 2. კონვოლუციური შრის ტიპური სტრუქტურა [3]

2. კონვოლუციური შრის ზოგადი სტრუქტურა

კონვოლუციური ნეირონული ქსელის ტიპური შრე შედგენილია სამი სტადიით [3], როგორც ეს მე-2 სურ-ზეა ნაჩვენები:

1. **წრფივი აქტივაციების კრებულის შექმნა** ერთი ან რამდენიმე პარალელური კონვოლუციის შესრულებით.

2. **დეტექტირება** – აქტივაციის არაწრფივი ფუნქციის გამოყენება ყველა წრფივი აქტივაციის მიმართ.

3. **სივრცითი გაერთიანება** (pooling) გამოსასვლელის შეცვლის მიზნით ქსელის მომდევნო შრეზე გადასაცემად.

თუ პირველ სტადიაზე გასაგებია, რომ სრულდება კონვოლუციის ოპერაცია, რომლის არსი აღწერილია ზემოთ, ხოლო მეორე სტადიაზე გამოიყენება აქტივაციის არაწრფივი ფუნქცია (ტიპური ფუნქციები განხილულია ლექციაში მრავალშრიანი სრულადბმული ქსელების შესახებ), მაშინ მესამე სტადიაზე ხორციელდება სივრცითი გაერთიანება.

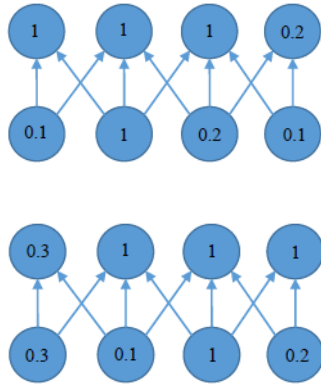
გაერთიანების არსი ისაა, რომ შეიცვალოს (ჩანაცვლდეს) ქსელის გამოსასვლელი კრებვითი სტატისტიკით გამოსასვლელთა მიდამოში.

გაერთიანების ოპერაციის მაგალითია max-pooling ოპერაცია, რომელიც უზრუნველყოფს მაქსიმუმის არჩევას მართკუთხა არეში.

გაერთიანების სხვა ფართოდ ცნობილ ოპერაციათა მაგალითებია საშუალო მართკუთხა მიდამოში (average pooling), L^2 – ნორმა მართკუთხა მიდამოში, შეწონილი საშუალო მანძილის საფუძველზე ცენტრალური პიქსელის მიმართ.

ფუნქციის არჩევისგან დამოუკიდებლად, სივრცითი გაერთიანება ხელს უწყობს იმას, რომ წარმოდგენა გახდეს ინვარიანტული შესასვლელთა ჩანაცვლების მიმართ. ეს ნიშნავს, რომ, თუ ობიექტი ოდნავ ჩანაცვლდება გამოსახულებაზე, მნიშვნელობები გაერთიანების სტადიის გამოსასვლელზე პრაქტიკულად არ შეიცვლება (სურ. 3).

მაგალითად, თუ წყდება გამოსახულებაზე სახის არსებობის განსაზღვრის ამოცანა, აუცილებელი არ არის თვალის ზუსტი ადგილმდგომარეობის ცოდნა გამოსახულებაზე, საკმარისია ინფორმაცია იმის შესახებ, რომ სახის მარცხენა და მარჯვენა მხარეზე არის თითო-თითო თვალი [3].



სურ. 3. max-pooling სივრცითი გაერთიანების ოპერაციის ინვარიანტობა წანაცვლების (მგრის) მიმართ. ოპერაცია სრულდება ერთეულოვანი ბიჯით [3]

ვინაიდან სივრცითი გაერთიანების ოპერაცია განაზოგადებს გამოძახილებს გარკვეულ მიდამოში, ამიტომ მოცემულ სტადიაზე შეიძლება ნეირონების ნაკლები რაოდენობის გამოყენება, ვიდრე დეტექტირების სტადიაზე.

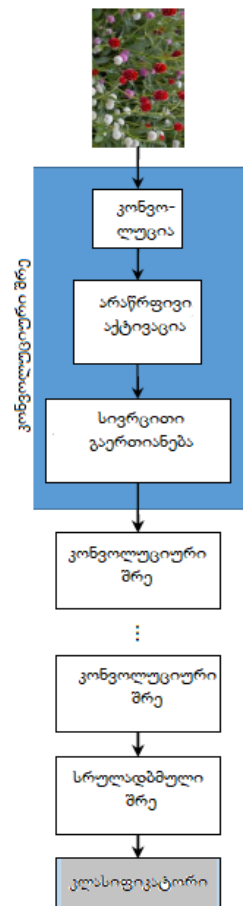
ამის რეალიზება ხდება მიდამოების გაერთიანებისას ერთზე უფრო დიდი ბიჯის (stride) გამოყენებით. შედეგად, იზრდება ქსელის გამოთვლითი ეფექტიანობა, რადგან შემდეგი ფენის შესასვლელის ზომა რამდენჯერმე ნაკლებია (დაახლოებით, stride-ჯერ), ვიდრე წინა კონვოლუციური შრის შემავალი შრე [3].

კონვოლუციური ნეირონული ქსელის ზოგადი სტრუქტურა წარმოდგენილია კონვოლუციურ შრეთა (კონვოლუციით, აქტივაციის ფუნქციით და სივრცითი გაერთიანებით შედგენილ სამეულთა)

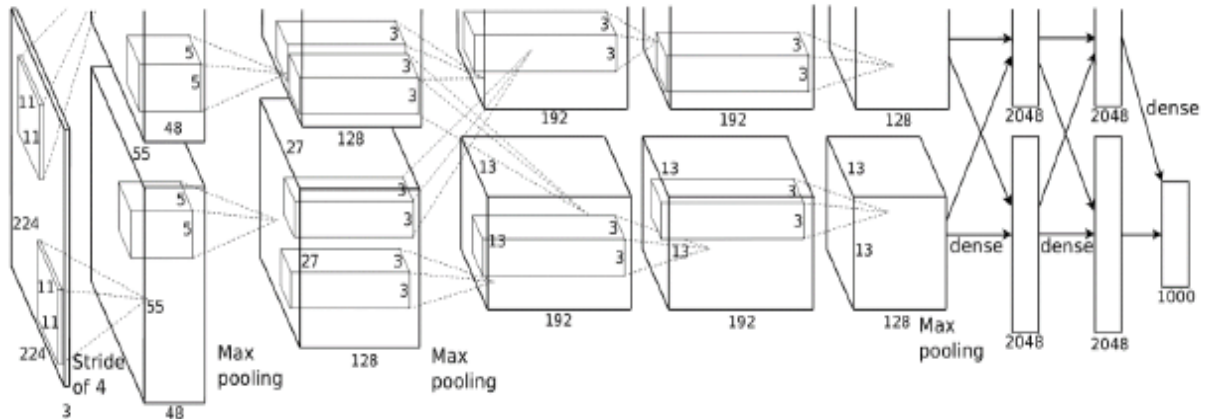
სტეკით, რომელსაც ბოლოში აქვს სრულადბმული ფენების გარკვეული რაოდენობა და კლასიფიკატორი (სურ. 4, 5).

რაც უფრო ზევითაა ფენა, გამოსახულებათა მით უფრო რთული ელემენტების გამოცნობის საშუალებას იძლევა იგი (წიბოების, წიბოების კომბინაციების, ტექსტურის, ობიექტების ნაწილების, ობიექტების კლასების), რაც გარკვეულ პირობებში უზრუნველყოფს ამოცანის ამონახსნის ხარისხის ზრდას.

გამოსახულებათა მრავალკლასიანი კლასიფიკაციის ამოცანის ამოხსნის შედეგების სტატისტიკა ImageNet (ცხრილი 1) მონაცემთა კრებულზე გვიჩვენებს, რომ კონვოლუციური ქსელების გაჩენის მომენტიდან შეინიშნება ქსელის სიღრმის ზრდა.



სურ. 4. კონვოლუციური ქსელის სტანდარტული სქემა



სურ. 5. AlexNet ქსელის სტრუქტურა [6]

ცხრილი 1

გამოსახულებათა მრავალკლასიანი კლასიფიკაციის ამოცანის ამოხსნისას უკეთესი შედეგების მომცემი კონვოლუციური ნეირონული ქსელების სიღრმის ზრდა

წელი	ჯგუფი	მეთოდი	კლასიფიკაციის შეცდომა	ფენათა რიცხვი
2012	SuperVision (University of Toronto, Canada)	Convolutional Neural Network (AlexNet)	0.15315	8
2013	Clarifai	Multiple Neural Networks	0.11197	8
2014	GoogLeNet	Convolutional Neural Network (GoogLeNet)	0.06656	22
2015	MSRA	Deep Residual Network (ResNet)	0.03567	152

3. კონვოლუციური ქსელის შემავალი მონაცემები მონაცემები, რომლებიც მიეწოდება კონვოლუციური ქსელის შესასვლელზე, როგორც წესი, მრავალარხიანია. ამასთან ერთად ყოველ არხს შეესა-

ბამება ხარისხობრივად სხვადასხვა დაკვირვება, რომლებიც მიღებულია ერთსა და იმავე წერტილზე სივრცეში ან დროში. ქვემოთ (ცხრილი 2) მოყვანილია შემავალი მონაცემების ტიპური მაგალითები.

ნეირონული ქსელების შემავალი მახასიათებლების მაგალითები [3] და კონვოლუციათა გამოთვლის ხერხი

	ერთარხიანი მონაცემები	მრავალარხიანი მონაცემები
1D	<i>აუდიოსიგნალი</i> – დისკრეტული სიგნალი, რომელიც მიღებულია გარკვეული ბიჯით დროში. კონვოლუცია გამოითვლება დროითი ღერძის გასწვრივ.	<i>ჩონჩხის ანიმაციის მონაცემები</i> . დროის ყოველ მომენტში პერსონაჟის პოზა აღიწერება კუთხეებით, რომლებიც შექმნილია ჩონჩხის სახსრების შესაბამის წერტილებზე. მონაცემების ყოველი არხი, რომელიც მიეწოდება კონვოლუციური ქსელის შესასვლელს, არის კუთხე ერთი სახსრის ღერძის გარშემო
2D	<i>წინასწარ დამუშავებული აუდიო სიგნალი</i> – სიგნალი ფურიეს დისკრეტული გარდაქმნის გამოყენების შემდეგ. არის ორგანზომილებიანი მატრიცა, რომელშიც სტრიქონები შეესაბამება სხვადასხვა სიხშირეს, სვეტები კი – სხვადასხვა წერტილს დროის მიხედვით. სიგნალის კონვოლუცია დროში იძლევა მოდელს, რომელიც ინვარიანტულია წანაცვლების მიმართ დროის მიხედვით, კონვოლუცია სიხშირეთა მიხედვით კი – სიხშირის მიმართ ინვარიანტულ მოდელს.	<i>ფერადი გამოსახულება</i> RGB (ან BGR) ფორმატში. კონვოლუციის ბირთვი მოძრაობს ჰორიზონტული და ვერტიკალური მიმართულებებით ერთდროულად და ამით უზრუნველყოფს ინვარიანტობას წანაცვლების ოპერაციის მიმართ.
3D	<i>სივრცითი მონაცემები</i> . ტიპური მაგალითია კომპიუტერული ტომოგრაფიის მონაცემები.	<i>ფერადი ვიდეო</i> - ორგანზომილებიანი ფერადი გამოსახულებების კრებული.

4. კონვოლუციური ქსელის გამომავალი მონაცემები

კონვოლუციური ქსელები მაღალი განზომილების გამომავალ მონაცემთა გენერირების საშუალებას იძლევა და არა მხოლოდ კლასის ჭდის ან ნამდვილი მნიშვნელობის წინასწარმეტყველების კლასიფიკაციის ან რეგრესიის აღდგენის ამოცანათა ამოხსნისას. როგორც წესი, ეს არის ობიექტი, რომელიც არის ქსელის სტანდარტული კონვოლუციური (ან სრულადბმული) ფენის გამოსასვლელზე მიღებულ ტენზორს. მაგალითად, მოდელი შეიძლება

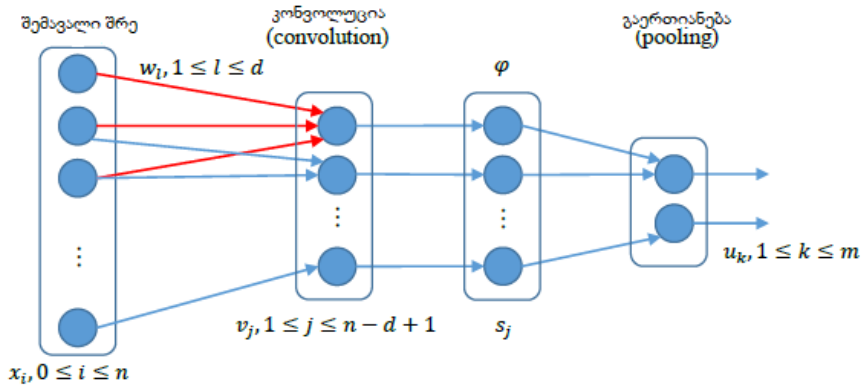
წარმოქმნიდეს სამგანზომილებიან ტენზორს $S_{i,j,k}$ ელემენტებით, რომლებიც შეესაბამება (i, j) პიქსელის k კლასისადმი მიკუთვნების ალბათობას. შედეგად მოდელი ყოველი პიქსელის მონიშვნისა და გამოსახულებაზე ობიექტების გამოყოფის – ესე იგი გამოსახულებათა სემანტიკური სეგმენტაციის ამოცანის გადაწყვეტის – საშუალებას იძლევა.

5. შეცდომის უკუგაგრძელების მეთოდი კონვოლუციური ნეირონული ქსელებისათვის

განვიხილოთ კონვოლუციური ნეირონული ქსელების სწავლების ზოგიერთი თავისებურება. შეც-

დომის უკუგავრცელების მეთოდის ზოგადი სქემა შედგება სამი ეტაპისგან: პირდაპირი გავლა (გრადიენტების გამოთვლა), შეცდომის ფუნქციისა და მისი გრადიენტის მნიშვნელობის გამოთვლა, უკუგავლა (შეწონვის კოეფიციენტების კორექცია). დავუშვათ, რომ კონვოლუციური ნეირონული ქსელის

სწავლების პროცესში არჩეულია შეცდომის $E(w)$ ფუნქცია. სიმარტივისათვის განვიხილოთ მხოლოდ ერთი ტიპური კონვოლუციური შრის შემცველი კონვოლუციური ქსელი. შემოვიღოთ აღნიშვნები ქვემოთ მე-6 სურათის შესაბამისად.



სურ. 6. ერთი კონვოლუციური შრის შემცველი ქსელის არქიტექტურა.
კონვოლუციის ბირთვის ზომა – d , კონვოლუციის გავლის ნაბიჯი – 1

ქსელის პირდაპირი გავლის ეტაპზე სრულდება წარმოებულების გამოთვლა უცნობი წონითი კოეფიციენტების მიხედვით. მოცემულ შემთხვევაში უცნობია კონვოლუციის წონები (შემავალი სიგნალის კონვოლუციის ბირთვის ზომა და შემოვლის ბიჯი ფიქსირებულია). შემავალი სიგნალისადმი კონვოლუციის გამოყენების შედეგი ასეთნაირად გამოიყურება:

$$v_j = \sum_{i=1}^d x_{j+i-1} w_i = w^T x_{j:j+d-1}.$$

მაშინ კონვოლუციის ფუნქციის წარმოებული ბირთვის წონითი კოეფიციენტებით გამოითვლება ქვემოთ მოცემული ფორმულით:

$$\frac{\partial v_j}{\partial w_i} = x_{j+i-1}, \quad \forall i = \overline{1, d}.$$

როგორც შედეგი, შეცდომის ფუნქციის წარმოებული კონვოლუციის ბირთვის წონებით განისაზღვრება როგორც

$$\frac{\partial E(w)}{\partial w_i} = \frac{\partial E}{\partial v} \frac{\partial v}{\partial w_i} = \sum_{j=1}^{n-d+1} \frac{\partial E}{\partial v_j} \frac{\partial v_j}{\partial w_i} = \sum_{j=1}^{n-d+1} \frac{\partial E}{\partial v_j} x_{j+i-1} = \langle \delta^{(conv)} * x \rangle(i) = \langle x * \delta^{(conv)} \rangle(i),$$

$$\delta^{(conv)} = \left(\frac{\partial E}{\partial v_j} \right)_{j=1, n-d+1},$$

$$\frac{\partial E}{\partial w} = \langle x * \delta^{(conv)} \rangle.$$

აქტივაციის ფუნქციისა და სივრცითი გაერთიანების (pooling) ფუნქციის წარმოებულები დამოკიდებულია არჩეულ ფუნქციათა სახეზე. აქტივაციის ფუნქციათა ჩამონათვალი განიხილებოდა ლექციაში

სრულადბმული ნეირონების შესახებ. გაერთიანების ფუნქციად შეიძლება იყოს არჩეული ნებისმიერი დიფერენცირებადი ნამდვილინა ფუნქცია. ზოგად შემთხვევაში სივრცითი გაერთიანების ფუნქცია არის q -ცვლადის ფუნქცია: $u_k = p(s_{k:k+q-1})$, სადაც q განსახილველად მოცემული მიდამოს ზომაა. გაერთიანების ფუნქციებს და მათ წარმოებულებს შეიძლება შემდეგი სახე გააჩნდეს:

$$p(x) = \begin{cases} \frac{\sum_{i=1}^q x_i}{q}, \quad \frac{\partial p}{\partial x_i} = \frac{1}{q}, \quad \text{average pooling} \\ \max_{i=1,q} x_i, \quad \frac{\partial p}{\partial x_i} = \begin{cases} 1, & x_i = \max_{i=1,q} x_i \\ 0, & \text{სხვა შემთხვევაში} \end{cases}, \quad \text{max-pooling} \\ \|x\|_p = \left(\sum_{i=1}^q |x_i|^p \right)^{\frac{1}{p}}, \quad \frac{\partial p}{\partial x_i} = \left(\sum_{i=1}^q |x_i|^p \right)^{\frac{1}{p}-1} |x_i|^{p-1}, \quad L_p\text{-pooling} \\ p: \mathbb{R}^q \rightarrow \mathbb{R}, \quad \exists \frac{\partial p}{\partial x_i} \end{cases} \quad \mathbb{R}$$

აქტივაციის ფუნქციისა და სივრცითი გაერთიანების ფუნქციის წარმოებულების არსებობისას მივიღებთ, რომ შეცდომის ფუნქციის წარმოებული გამოითვლება შემდეგნაირად:

$$\frac{\partial E(w)}{\partial w_i} = \sum_{j=1}^{n-d+1} \frac{\partial E}{\partial u_k} \frac{\partial p}{\partial s_j} \frac{\partial \varphi}{\partial v_j} x_{j+i-1} = \langle x * \delta^{(conv)} \rangle(x_i),$$

$$\delta_{j,k}^{(conv)} = \frac{\partial E}{\partial u_k} \frac{\partial p}{\partial s_j} \frac{\partial \varphi}{\partial v_j} = \delta_k^{(pool)} \frac{\partial p}{\partial s_j} \frac{\partial \varphi}{\partial v_j}, \quad \delta_k^{(pool)} = \frac{\partial E}{\partial u_k}.$$

შემდეგ შეცდომის ფუნქციისა და მისი წარმოებულების მიღებული მნიშვნელობების საფუძველზე ხორციელდება წონითი კოეფიციენტების კორექცია ქსელის უკუგავლის მეშვეობით. გაერთიანების არჩევისას მაქსიმუმის მიხედვით ცხადია, რომ შეცდომის ფუნქციის გრადიენტი უკანასკნელ შრეზე მიდის s_j ნეირონისკენ, რომლის გამოსავლელზე მიღებულია მაქსიმალური მნიშვნელობა. ზოგად შემთხვევაში იგი ვრცელდება $\delta_k^{(pool)}$, $1 \leq k \leq m$ მნიშვნელობის შესაბამისად. მერე ეს

მნიშვნელობა გადაეცემა შექცეული მიმართულე-ბით და გრადიენტების მნიშვნელობა კონვოლუციურ შრეზე განისაზღვრება $\delta_{j,k}^{(conv)}$, $1 \leq j \leq n-d+1$ სიდიდის გამოსათვლელი ფორმულის შესაბამისად [7]. მიღებული გრადიენტების საფუძველზე ხდება კონვოლუციის ბირთვის წონათა კორექტირება.

შეცდომის უკუგავრცელების მეთოდის მოყვანილი აღწერა ერთი კონვოლუციური შრის შემცველი ქსელის შემთხვევისათვის შეიძლება იყოს განზოგადებული კონვოლუციურ შრეთა ნებისმიერ რაოდენობაზე.

6. დასასწავლ პარამეტრთა რაოდენობის განსაზღვრა. ქსელის შესანახად აუცილებელი მეხსიერების მოცულობის შეფასება

კონვოლუციური ქსელის დასასწავლ პარამეტრთა რაოდენობის განსაზღვრის და ქსელის შესანახად აუცილებელი მეხსიერების მოცულობის შეფასების ამოცანები საყურადღებოც არის და პრაქტიკულად მნიშვნელოვანაც. პირველი ამოცანის გადაჭრა საშუალებას იძლევა შეფასდეს პარამეტრთა იმ სივრცის განზომილება, რომელშიც წყდება მიზნობრივი ფუნქციის მინიმიზაციის ამოცანა. მეორე ამოცანის გადაჭრა კი იმ გამოთვლითი რესურსების შეფასების საშუალებას იძლევა, რომლებიც აუცილებელია ნეირონული ქსელის სწავლების ან ტესტირების ჩასატარებლად.

კონვოლუციური ნეირონული ქსელის დასასწავლ პარამეტრთა რიცხვი განისაზღვრება კონვოლუციურ შრეთა რაოდენობით და კონვოლუციური ბირთვების განზომილებით. ვინაიდან ეს სტატია აგებულია ღრმა სწავლების გამოყენების მაგალითებზე კომპიუტერული მხედველობის ამოცანათა გადაწყვეტისას, დავუშვათ, რომ ქსელის შესასვ-

ლელზე არის $w \times h$ გარჩევითობის ერთარხიანი გამოსახულება. ამ დროს ქსელი შეიცავს N კონვოლუციურ შრეს (კონვოლუციით, აქტივაციის ფუნქციის გამოთვლით და სივრცითი გაერთიანებით შედგენილ სამეულს). თითოეულ შრეზე ხორციელდება ნიშანთა რუკის კონვოლუცია ფილტრით, რომლის ბირთვის ზომა $w_c \times h_c$ სიდიდეს შეადგენს, სადაც $1 \leq i \leq N - 1$ შრის ნომერია. გამოსახულების შემოვლას ახორციელებს ფილტრი გარკვეული s_c ნაბიჯით. მაშინ ქსელის დასასწავლ პარამეტრთა რაოდენობა $\sum_{i=1}^N w_c \cdot h_c$ სიდიდეს შეადგენს. ზოგად შემთხვევაში ყოველ შრეზე შეიძლება გამოითვლებოდეს კონვოლუცია არა ერთი ბირთვით, არამედ ერთნაირი ზომის რამდენიმე ბირთვით. იმ დაშვებით, რომ i ნომრით სრულდება k_i კონვოლუცია, პარამეტრების ჯამური რაოდენობა $\sum_{i=1}^N k_i \cdot w_c \cdot h_c$ სიდიდეს შეადგენს.

შევაფასოთ მეხსიერების მოცულობა, რომელიც აუცილებელია ქსელის პირველი კონვოლუციური შრის შესანახად. ქსელის შემავალი შრე შეიცავს $w \times h$ პიქსელს. k_1 კონვოლუციის გამოყენება შემავალი შრის მიმართ $k_1 \times \left(\left\lfloor \frac{w - w_{c_1}}{s_{c_1}} \right\rfloor + 1 \right) \times \left(\left\lfloor \frac{h - h_{c_1}}{s_{c_1}} \right\rfloor + 1 \right)$ განზომილების ნიშანთა რუკის ფორმირებას განაპირობებს, აქტივაციის ფუნქციის გამოყენება მიღებული რუკის თითოეული ელემენტის მიმართ იძლევა ნიშნების რუკას იდენტური $k_1 \times \left(\left\lfloor \frac{w - w_{c_1}}{s_{c_1}} \right\rfloor + 1 \right) \times \left(\left\lfloor \frac{h - h_{c_1}}{s_{c_1}} \right\rfloor + 1 \right)$ განზომილებით. შემოვიღოთ აღნიშვნები:

$$w_1 = \left\lfloor \frac{w - w_{c_1}}{s_{c_1}} \right\rfloor + 1, \quad h_1 = \left\lfloor \frac{h - h_{c_1}}{s_{c_1}} \right\rfloor + 1.$$

დავუშვათ, რომ სივრცითი გაერთიანების ოპერაცია გამოიყენება აქტივაციის ფუნქციის გამოსახულებაზე მიღებული ნიშანთა რუკის ყოველი არხის მიმართ, განიხილება $w_{p_1} \times h_{p_1}$ ზომის მიდამო და გავლა ხორციელდება s_{p_1} ნაბიჯით. მაშინ ნიშანთა რუკის განზომილება აიგება რუკის იმ განზომილების ანალოგიურად, რომელიც მიღებული იყო კონვოლუციის ოპერაციის გამოყენების შემდეგ:

$$k_1 \times \left(\left\lfloor \frac{w_1 - w_{p_1}}{s_{p_1}} \right\rfloor + 1 \right) \times \left(\left\lfloor \frac{h_1 - h_{p_1}}{s_{p_1}} \right\rfloor + 1 \right),$$

როგორც შედეგი, მეხსიერების ის მოცულობა, რომელიც აუცილებელია პირველი კონვოლუციური შრის შესანახად, განისაზღვრება ფორმულით:

$$\left(2k_1 w_1 h_1 + k_1 \left(\left\lfloor \frac{w_1 - w_{p_1}}{s_{p_1}} \right\rfloor + 1 \right) \left(\left\lfloor \frac{h_1 - h_{p_1}}{s_{p_1}} \right\rfloor + 1 \right) \right) \cdot \text{sizeof}(\text{type}),$$

სადაც $\text{sizeof}(\text{type})$ – ბაიტების რაოდენობაა, რომელიც საჭიროა ერთი ნეირონის გამოძახილის შენახვისათვის. თუ გვაქვს ანალოგიურად მოწყობილი N კონვოლუციური შრე, მაშინ მთელი ქსელის შესანახად საჭიროა მეხსიერების შემდეგი მოცულობა:

$$\left(w \cdot h + \sum_{i=1}^N \left(2k_i w_i h_i + k_i \left(\left\lfloor \frac{w_i - w_{p_i}}{s_{p_i}} \right\rfloor + 1 \right) \left(\left\lfloor \frac{h_i - h_{p_i}}{s_{p_i}} \right\rfloor + 1 \right) \right) \right) \cdot \text{sizeof}(\text{type}),$$

$$w_i = \left\lfloor \frac{w_{i-1} - w_{c_i}}{s_{c_i}} \right\rfloor + 1,$$

$$h_i = \left\lfloor \frac{h_{i-1} - h_{c_i}}{s_{c_i}} \right\rfloor + 1, \quad w_0 = w, \quad h_0 = h.$$

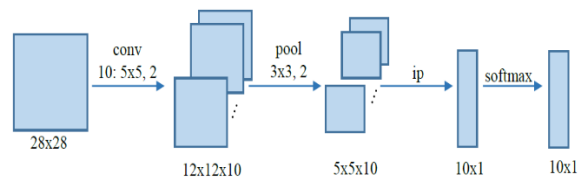
უნდა აღინიშნოს, რომ განხილული მაგალითი არის კონვოლუციური ქსელის კერძო შემთხვევა და გვიჩვენებს ასეთი ქსელის შესანახად აუცილებელი მეხსიერების მოცულობის შეფასების მხოლოდ ზო-

გად პროცედურას. ასე, მაგალითად, ქსელის შესასვლელზე შეიძლება იყოს სიგნალი, რომელიც სტრუქტურის მიხედვით განსხვავდება განხილულისგან (ვთქვათ, სამარხიანი გამოსახულება). როგორც ამის შედეგი, იცვლება კონვოლუციური ბირთვების განზომილებები. კონვოლუციის ოპერაციის შესრულებისას ნიშნების შემავალ რუკას შეიძლება ემატებოდეს მინდვრები რუკის განზომილების შესანარჩუნებლად გამოსასვლელზე. კონვოლუციური ქსელი კონვოლუციურ შრეებთან ერთად შეიძლება შეიცავდეს სრულადბმულ შრეებს. გარდა ამისა, სწავლების პროცესში ქსელის შესასვლელზე შეიძლება მიეწოდებოდეს გამოსახულება ფიქსირებული ზომის მთელი კრებულის, შეკვრის – პაკეტის (batch) სახით გამოთვლათა ეფექტურობისა და ქსელის დასწავლის სიჩქარის გაზრდის მიზნით. აუცილებელი მეხსიერების წინასწარი შეფასება განსაზღვრავს სწავლების პროცედურის შესრულების შესაძლებლობას არსებულ გამოთვლით რესურსებზე და გამოსახულებათა დასამუშავებელი პაკეტის მისაღები ზომის შერჩევის საშუალებას იძლევა. ამ პროცედურის აუცილებლობა, უმთავრესად, განპირობებულია იმით, რომ სწავლების პროცესში გამოიყენება გრაფიკული პროცესორები მეხსიერების შეზღუდული მოცულობით.

7. ხელნაწერი ციფრების კლასიფიკაციის ამოცანაში კონვოლუციური ნეირონული ქსელების გამოყენების მაგალითი

კონვოლუციური ქსელების გამოყენების მაგალითად განვიხილოთ ერთშიანი კონვოლუციური ნეირონული ქსელი ხელნაწერი ციფრების კლასიფიკაციის ამოცანაში (სურ. 7).

1. კონვოლუციური შრე. შეიცავს 10 ფილტრს: ზომა - 5×5 ელემენტი, შემოვლის ბიჯი - 2.
2. სივრცითი გაერთიანების (მაქსიმუმის ოპერაციით) შრე. შეიცავს 10 ფილტრს: ზომა - 3×3 ელემენტი, შემოვლის ბიჯი - 2.
3. სრულადბმული შრე. შეიცავს 10 გამომავალ ელემენტს (კლასების რიცხვის შესაბამისად მოცემულ ამოცანაში).
4. აქტივაციის softmax ფუნქცია.



სურ. 7. ერთშიანი კონვოლუციური ქსელის სტრუქტურა

ქვემოთ მოტანილია ნეირონული ქსელის მოცემული არქიტექტურის აღწერა (ლისტინგი 1) და სწავლების მეთოდის პარამეტრები (ლისტინგი 2) prototxt ფორმატში, რომელსაც იყენებს ბიბლიოთეკა Caffe [14].

<pre> name: "SimpleCNN" layer { name: "mnist" type: "ImageData" top: "data" top: "label" include { phase: TRAIN } transform_param { scale: 0.00390625 } image_data_param { source: "mnist_dataset/train.lst" new_width: 28 new_height: 28 batch_size: 100 root_folder: "mnist_dataset/train/" } } layer { name: "mnist" type: "ImageData" top: "data" top: "label" include { </pre>	<pre> phase: TEST } transform_param { scale: 0.00390625 } image_data_param { source: "mnist_dataset/t10k.lst" new_width: 28 new_height: 28 batch_size: 100 root_folder: "mnist_dataset/t10k/" } } layer { name: "conv" type: "Convolution" bottom: "data" top: "conv" convolution_param { num_output: 10 kernel_size: 5 stride: 2 weight_filler { type: "gaussian" std: 0.01 } } </pre>
<pre> } } layer { name: "pool" type: "Pooling" </pre>	<pre> } } layer { name: "loss" </pre>

<pre> bottom: "conv" top: "pool" pooling_param { pool: MAX kernel_size: 3 stride: 2 } } layer { name: "ip" type: "InnerProduct" bottom: "pool" top: "ip" inner_product_param { num_output: 10 </pre>	<pre> type: "SoftmaxWithLoss" bottom: "ip" bottom: "label" top: "loss" } layer { name: "accuracy" type: "Accuracy" bottom: "ip" bottom: "label" top: "accuracy" include { phase: TEST } } </pre>
--	--

ლისტინგი 1. კონვოლუციური ქსელის არქიტექტურის აღწერა Caffe ბიბლიოთეკის გამოყენებით

<pre> net:"mnist_cnn.prototxt" test_iter: 100 test_interval: 500 base_lr: 0.01 momentum: 0.9 weight_decay:0.0005 lr_policy: "inv" gamma: 0.0001 power: 0.75 display: 100 max_iter: 10000 snapshot: 5000 snapshot_prefix: "mnist_cnn/mnist_cnn" solver_mode: CPU </pre>
--

ლისტინგი 2. კონვოლუციური ქსელის სწავლების პარამეტრები

მე-3 ცხრილში მოცემულია ხელნაწერი ციფრების კლასიფიკაციის სიზუსტის მნიშვნელობები მონაცემთა MNIST კრებულზე. ისინი მიღებულია ისეთი ქსელის გამოყენებით, რომელიც შეესაბამება ლოგისტიკურ რეგრესიას და ერთშრიან კონვოლუციურ ქსელს.

ცხრილი 3

მონაცემთა MNIST კრებულზე ხელნაწერი ციფრების კლასიფიკაციის შედეგები

მეთოდი	სიზუსტე, %
ლოგიკური რეგრესია	90.6
ერთშრიანი კონვოლუციური ქსელი	97.6

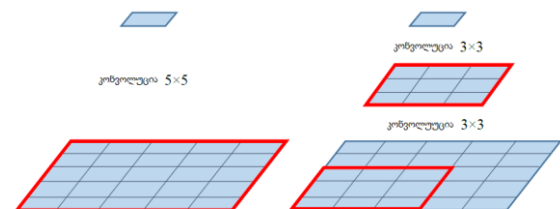
8. კონვოლუციური ქსელების აგების პრინციპები

განვიხილოთ კონვოლუციური ქსელების აგების ზოგიერთი პრინციპი (რეკომენდაცია).

- შემავალი მონაცემების წინასწარი დამუშავების განხორციელება.** საწვრთნელი მონაცემების წინასწარი დამუშავება კონვოლუციური ქსელის ეფექტურობისა და სწავლების პროცედურის დაჩქარების საშუალებას იძლევა. წინასწარი დამუშავების ტიპური მაგალითია საწვრთნელი სიმრავლის ყველა გამოსახულებით მიღებული საშუალო გამოსახულების გამოკლება.
- ქსელის წარმოდგენაში «ვიწრო ყელის» გაჩენის არიდება, განსაკუთრებით საწყის ფენებში.** ქსელის არქიტექტურის აგებისას აზრი აქვს თავიდან ავიცილოთ ინფორმაციის ექსტრემალური შეკუმშვა. ზოგად შემთხვევაში წარმოდგენის განზომილება უნდა მცირდებოდეს თანდათანობით შემავალი მონაცემებიდან ქსელის გამოსასვლელამდე, ესე იგი ისეთი წარმოდგენის მიღებამდე, რომელიც გამოიყენება ამოცანის ამო-

სახსნელად. წარმოდგენის განზომილება იძლევა ინფორმაციული კონტენტის მიახლოებით შეფასებას [8].

- დიდი განზომილების კონვოლუციური ფენების ჩანაცვლების განხორციელება უფრო მცირე განზომილების კონვოლუციითა სტეკით [8,9].** დიდი (5×5 , 7×7 და უფრო მეტი) განზომილების ფილტრების გამოყენება უფრო მეტ მეზობელ სიგნალს შორის დამოკიდებულებაზე ხელის დავლების საშუალებას იძლევა, მაგრამ გამოთვლითი სისტემის თვალსაზრისით ძვირადღირებული ოპერაცია შეიძლება იყოს. ამის მიუხედავად მაღალი განზომილების ფილტრის გამოყენება შეიძლება დავიყვანოთ უფრო დაბალი განზომილების ფილტრების მიმდევრობით გამოყენებამდე. ასე, მაგალითად, კონვოლუცია 5×5 ფილტრით შეიძლება შეიცვალოს ორი მიმდევრობითი კონვოლუციით 3×3 ზომის ფილტრებით (სურ. 8).



სურ. 8. კონვოლუციის ფილტრით 5×5 ჩანაცვლება ორი მიმდევრობითი კონვოლუციით 3×3 ზომის ფილტრებით

ამ დროს ხდება ქსელის ფორმირება პარამეტრების ნაკლები რიცხვით, მაგრამ იმავე ზომის შესასვლელით და გამოსასვლელის სიღრმით.

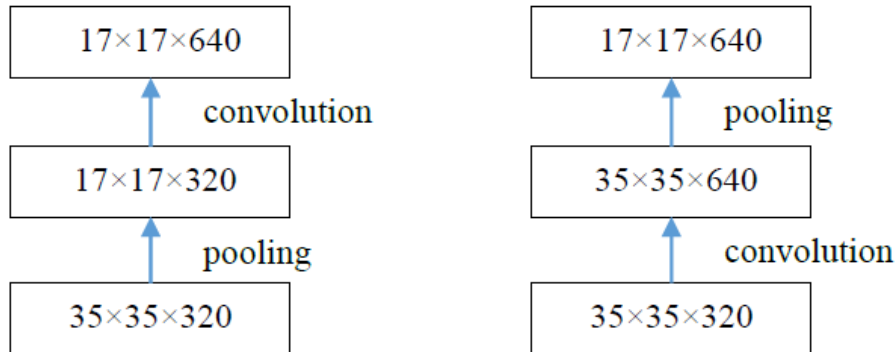
მოტანილი მაგალითიდან შეიძლება მოგვეჩვენოს, რომ კონვოლუციები 3×3 ზომაზე ნაკლები

ბირთვებით უსარგებლოა. სინამდვილეში კი ეს ასე არ არის.

კონვოლუცია 3×3 ბირთვით შეიძლება გაიშალოს 2×2 ან 3×1 და 1×3 ზომების ორი კონვოლუციის მიმდევრობად, მაგრამ ასეთი ოპერაციის ჩატარებისას მოგება პარამეტრთა და ჩასატარებელ გამოთვლათა რიცხვის მიხედვით არ არის ისეთი მნიშვნელოვანი, როგორც წინა მაგალითში.

ცხადია, რომ კონვოლუცია $m \times m$ ბირთვით შეიძლება დავიყვანოთ კონვოლუციათა $m \times 1$ და $1 \times m$ წყვილის გამოყენებამდე, მაგრამ პრაქტიკა აჩვენებს, რომ დიდი განზომილების ქსელის ადრეულ შრეებზე ასეთი გარდასახვის გამოყენება არაეფექტურია. ამ დროს საშუალო (12-დან 20-მდე) განზომილების ნიშანთა რუკაზე ადგილი აქვს საწინააღმდეგო ეფექტს [8].

4. სივრცითი აგრეგაცია უნდა ტარდებოდეს უფრო დაბალი განზომილების ნიშანთა რუკებით გამოთვლითი სირთულის შესამცირებლად. კონვოლუციის განხორციელების წინ შეიძლება შემცირდეს შემავალი წარმოდგენის ზომა ყოველგვარი გვერდითი ეფექტის გარეშე [8]. რეალიზება ხდება სივრცითი გაერთიანების (pooling) ან inception-მოდულების შემოტანით (სურ. 9). გაერთიანების ოპერაციის გამოყენება შემავალი შრის მიმართ (სურ. 9, მარცხენა დიაგრამა) იწვევს ინფორმაციის დაკარგვას საწყისი წარმოდგენის შესახებ, რაც მეტყველებს «ვიწრო ყელის» არსებობაზე ქსელში. კონვოლუციის განხორციელება სრული გარჩევითობის პირობებში (სურ. 9, მარჯვენა დიაგრამა) წარმოქმნის გამოთვლით სირთულეს.

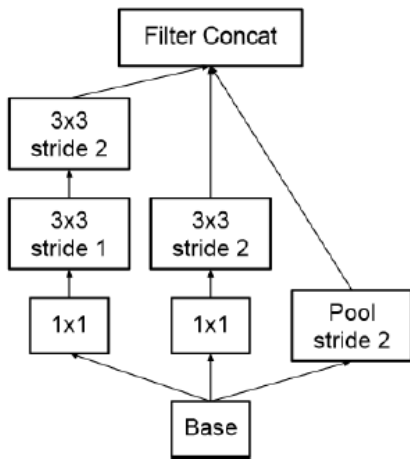


სურ. 9. ნიშანთა რუკის განზომილების შემცირების ორი ალტერნატიული გზა (შემავალი რუკის ზომა $35 \times 35 \times 320$, გამომავალი რუკის ზომა $17 \times 17 \times 640$) [8]

inception-მოდულის შემოტანით (სურ. 10) ორივე ამოცანის გადაჭრა ხორციელდება. მოდულის გამოსასვლელზე ხდება იმ ნიშნების კონკატენაცია, რომლებიც მიღებულია ნიშანთა საწყისი რუკის სრული გარჩევითობის პირობებ-

ში (ე.ი. მიღებულია კონვოლუციების გამოთვლისას) და რუკის ნაკლები გარჩევითობის დროს აგებული ნიშნები (რომლებიც მიღებულია სივრცითი გაერთიანების შედეგად).

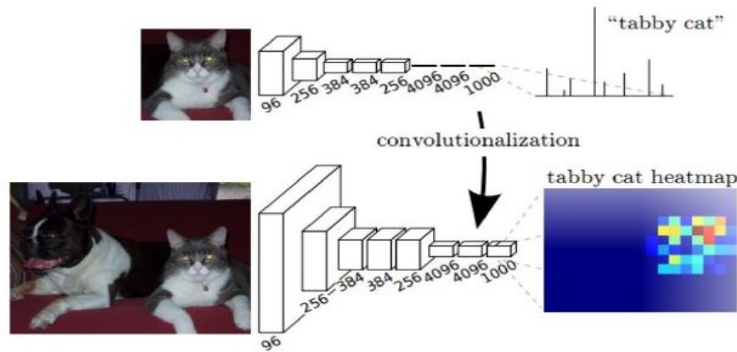
5. ქსელის სიღრმისა და სიგანის დაბალანსება. ქსელის ოპტიმალური მწარმოებლურობა შეიძლება იყოს მიღწეული ყოველ კონვოლუციურ შრეზე ფილტრების რაოდენობის და ქსელის სიღრმის დაბალანსების გზით. ქსელის სიგანისა და სიღრმის გაუმჯობესებამ შეიძლება ხელი შეუწყოს უფრო მაღალი ხარისხის ქსელების შექმნას. მაგრამ ოპტიმალური გაუმჯობესება მაშინ შეიძლება იყოს მიღწეული, როცა ორივე პარამეტრი პარალელურად იზრდება [8].



სურ. 10. inception-მოდულის ზოგადი სტრუქტურა [8]

6. ტარდება ქსელის უკანასკნელ სრულადბმულ შრეთა ჩანაცვლება კონვოლუციური შრეებით ნებისმიერი ზომის გამოსახულებებთან მუშაობის შესაძლებლობის უზრუნველსაყოფად [11]. სრულადბმული შრის შესაცვლელად კონვოლუციურით ნიშანთა შემავალი $c_1 \times w \times h$ ზომის რუკის მიმართ შემოთავაზებულია მოცემული რუკის $c_1 \times w \times h$ ზომის მქონე c_2 ფილტრის გამოყენება. შედეგად მიიღება ნიშანთა $c_2 \times 1 \times 1$ განზომილების ვექტორი. პარამეტრთა რაოდენობის შემცირების გარდა სრულადბმულ შრეთა

შეცვლისას, ასეთი გარდაქმნა ქსელის შესასვლელზე სხვადასხვა ზომის შემავალ მონაცემებთან მუშაობის საშუალებას იძლევა, რაც ფართოდ გამოიყენება მრავალი პრობლემის (კერძოდ, ობიექტების დეტექტირებისა და სემანტიკური სეგმენტაციის ამოცანის) გადაჭრისას. განვიხილოთ მაგალითი (სურ. 11). დავუშვათ, რომ წყდება გამოსახულებათა კლასიფიკაციის ამოცანა, ქსელის შესასვლელზე მიეწოდება ფიქსირებული ზომის სურათი, გამოსასვლელზე ხდება იმის უტყუარობათა ვექტორის ფორმირება, რომ შემავალი გამოსახულება მიეკუთვნება კლასებიდან თითოეულს [11]. ტესტირების ეტაპზე შევცვალოთ მოცემულ ქსელში სრულადბმული შრეები კონვოლუციური ფენებით, ხოლო შესასვლელზე მივაწოდოთ უფრო დიდი გარჩევითობის გამოსახულება. მაშინ ქსელის ყოველი შრის განზომილება დამოკიდებული არ არის შემავალი მონაცემების ზომაზე, ხოლო გამოსასვლელზე უტყუარობათა ერთი ვექტორის ნაცვლად ხდება ვექტორთა მთელი კრებულის ფორმირება. გამომავალ ვექტორთა რაოდენობა დამოკიდებულია კონვოლუციური ფილტრების შესაძლო განლაგებათა რიცხვზე ქსელის პირველი შრისათვის. ამრიგად, არაცხადი სახით, ხდება «მორბენალი ფანჯრის» (sliding window) მეთოდის რეალიზება. ამ ფანჯრის ზომა შეესაბამება ქსელის შესასვლელის ზომას. სემანტიკური სეგმენტაციის ამოცანის გადაჭრისას მდგომარეობა უფრო რთულია, ვინაიდან აუცილებელია გამოსასვლელის მასშტაბირება საწყისი გამოსახულების ზომამდე.



სურ. 11. კლასიფიკაციის ამოცანიდან დეტექტირების ამოცანაზე გადასვლა სრულადბმულ შრეთა ჩანაცვლებით კონვოლუციური შრეებით [11]

9. მოდელის დეგრადაციის პრობლემა. ღრმა ნარჩენი ქსელები

ღრმა ნეირონული ქსელების სწავლების პროცესში შეიძლება ფართოდ ცნობილი **დეგრადაციის პრობლემა** ვიხილოთ. იგი შემდეგნაირად ვლინდება:

ქსელის სიღრმის გაზრდისას ხდება სიზუსტის გაჯერება და შემდეგ იგი სწრაფად მცირდება (დეგრადირებს). მოცემული პრობლემა არ არის მოდელის ზედმეტად სწავლების შედეგი და დამატებითი შრეების შეტანას მივყავართ წვრთნის შეცდომის კიდევ უფრო დიდ მნიშვნელობამდე. სწავლების სიზუსტის დეგრადაცია მიუთითებს იმაზე, რომ ყველა ღრმა მოდელი ერთნაირად ადვილად არ ემორჩილება ოპტიმიზაციას იმის გამო, რომ ჩნდება გრადიენტების სწრაფი მილევა (vanishing gradient problem) ქსელის შერბულებული მიმართულებით გავლისა და წონების კორექციის პროცესში.

ღრმა ნარჩენი ქსელები (Deep residual Networks) გამიზნულია დეგრადაციის პრობლემის გადასაჭრელად. ნაცვლად იმ დაშვებისა, რომ ქსელის ფენათა გარკვეული თანამიმდევრობა პირდაპირ ახდენს საბაზო ასახვის აპროქსიმაციას, თვლიან, რომ ეს ფენები ახდენს ნარჩენი ასახვის აპროქსიმაციას.

თუ ფორმალურად საბაზო ასახვა აღინიშნება როგორც $H(x)$, მაშინ ფენების მოცემული კრებული-სათვის მიიღება, რომ ფენების აპროქსიმაცია ხდება $F(x) = H(x) - x$ ასახვით, ხოლო საბაზო ასახვა შეიძლება წარმოვადგინოთ $F(x) + x$ ფორმით, რაც ნიშნავს ნიშანთა რუკების შეკრებას ყოველი ელემენტის მიხედვით. თვლიან, რომ ნარჩენი ასახვის ოპტიმიზაცია საბაზო ასახვის ოპტიმიზაციაზე უფრო ადვილია, ვინაიდან დიფერენცირებისას ასეთი ფუნქციის გრადიენტი არ უდრის ნულს. უკიდურეს შემთხვევაში, თუ იგივეური $H(x) = x$ გარდაქმნა არის ოპტიმალური, მაშინ ნარჩენის დაყვანა ნულამდე გაცილებით იოლია, ვიდრე იგივეური ასახვის აპროქსიმაცია არაწრფივი შრეების თანამიმდევრობით. $F(x) + x$ ასახვა შეიძლება წარმოვადგინოთ სწრაფი ბმების (shortcut connection) მქონე პირდაპირი გავრცელების ქსელით [12] (სურ. 12).

ქვემოთ ნაჩვენებია ორშრიანი ბლოკისათვის (სურ. 12) გარდაქმნას შემდეგი სახე აქვს:

$$y = F(x, W_i) + x = W_2 \phi(W_1 x) + x,$$

$\phi(\cdot)$ – აქტივაციის ReLU (Rectified Linear Unit)

ფუნქციაა (ნახევრადწრფივი ელემენტი):

$$\varphi(z) = \begin{cases} 0, & \text{თუ } z < 0 \\ z, & \text{თუ } z \geq 0 \end{cases}$$

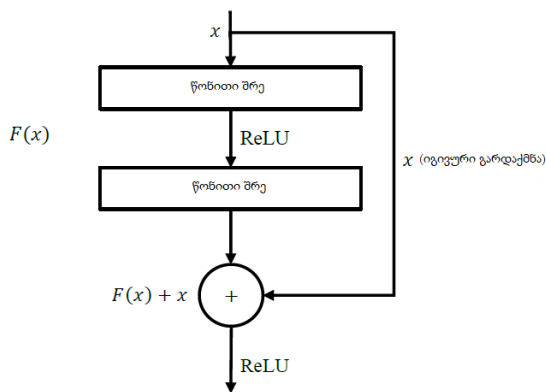
ზოგად შემთხვევაში $F(x, W_i)$ და x სიდიდეებს აქვს სხვადასხვა განზომილება. ამ ვითარების გამოსასწორებლად საკმარისია შესრულდეს ნიშნების შემავალი ვექტორის პროექცია. შედეგად, გარდასახვა შეიძლება შემდეგნაირად ჩაიწეროს:

$$y = F(x, W_i) + W_s x,$$

სადაც W_s – პროექციების მატრიცაა.

ნარჩენი $F(\cdot)$ ფუნქციის სახე საკმაოდ მოქნილია, იგი შეიძლება შეესაბამებოდეს ორი ან მეტი შრით შედგენილ ბლოკს.

თუ მოცემულ ფუნქციას შეესაბამება მხოლოდ ერთი ფენა, მაშინ გარდაქმნა $y = Wx + x$ წრფივი ფენის ეკვივალენტურია, რომლისთვისაც, [12]-ში ჩატარებული ექსპერიმენტების თანახმად, მოგება არ გამოვლინდა.



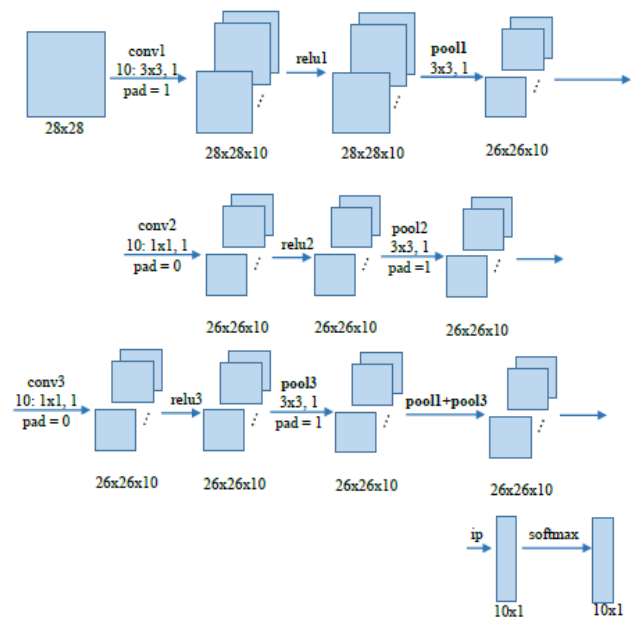
სურ. 12 ღრმა ნარჩენი ქსელების სტრუქტურული ბლოკი [12]

10. ღრმა ნარჩენი ქსელების გამოყენების მაგალითი ხელნაწერ ციფრთა კლასიფიკაციის ამოცანაში

შევჩერდეთ ისეთი ღრმა ნარჩენი ქსელის უმარტივეს ამოცანაზე, რომელიც შეიცავს ერთ ტიპურ სტრუქტურულ ბლოკს.

ქვემოთ (ნახ. 13) მოტანილია ასეთი ქსელის არქიტექტურა ხელნაწერი ციფრების კლასიფიკაციის ამოცანისათვის.

ადვილი შესამჩნევია, რომ ქსელის აგებისას (ე.ი. შრეთა პარამეტრებისათვის მნიშვნელობების მინიჭებისას) უნდა ხდებოდეს ნიშნთა რუკების ზომათა თანხვედრის აუცილებლობის გათვალისწინება ოპერაციის კორექტული შესრულებისათვის ელემენტიდან ელემენტზე გადასვლისას.



სურ. 13 უმარტივესი ნარჩენი ქსელის სტრუქტურა

ექსპერიმენტების შედეგები (ცხრილი 4) მონაცემთა MNIST კრებულზე ასახავს კლასიფიკაციის დაახლოებით იმავე სიზუსტეს, რომელთანაც საქმე გვექონდა ადრე ერთშრიანი კონვოლუციური ნეირონული ქსელისათვის. ეს ფაქტი აიხსნება ამოცანის მცირე განზომილებით. ნარჩენი ბლოკის გამოყენება კლასიფიკაციის სიზუსტის მნიშვნელოვანი შემცირების საშუალებას არ იძლევა.

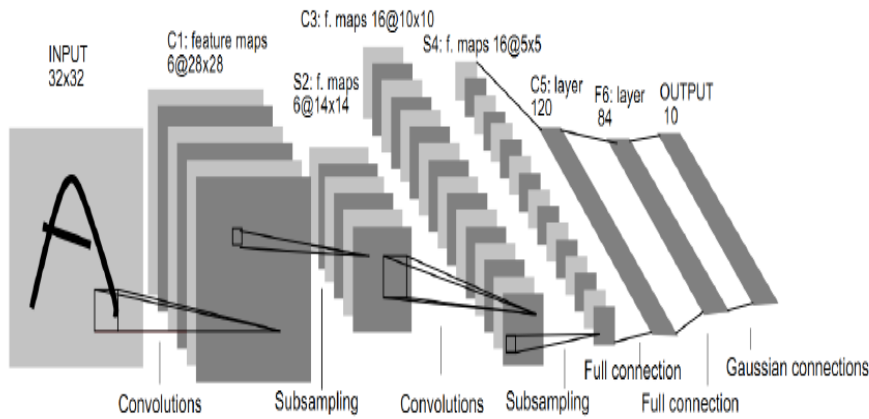
ცხრილი 4

11. კონვოლუციური ნეირონული ქსელების არქიტექტურათა განვითარება

მონაცემთა MNIST კრებულზე ხელნაწერი ციფრების კლასიფიკაციის შედეგები

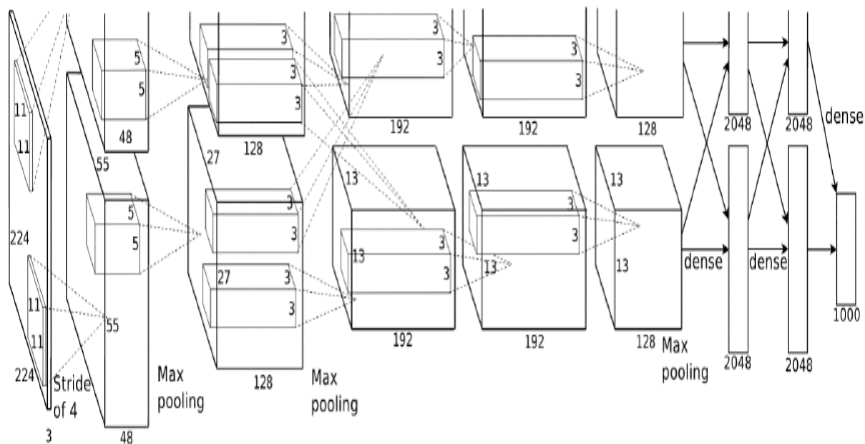
მეთოდი	სიზუსტე, %
ლოგიკური რეგრესია	90.6
ერთშრიანი კონვოლუციური ქსელი	97.6
ნარჩენი ქსელი ერთი ტიპური ბლოკით	97.7

კონვოლუციური ნეირონული ქსელების ერთ-ერთი ყველაზე პირველად შემოთავაზებულ - არქიტექტურა არის LeNet-5. მისი დანიშნულებაა ხელნაწერ სიმბოლოთა გამოცნობის ამოცანის გადაჭრა. იგი შედგენილია ორი კონვოლუციური და ორი სრულადბმული შრით (სურ. 14).



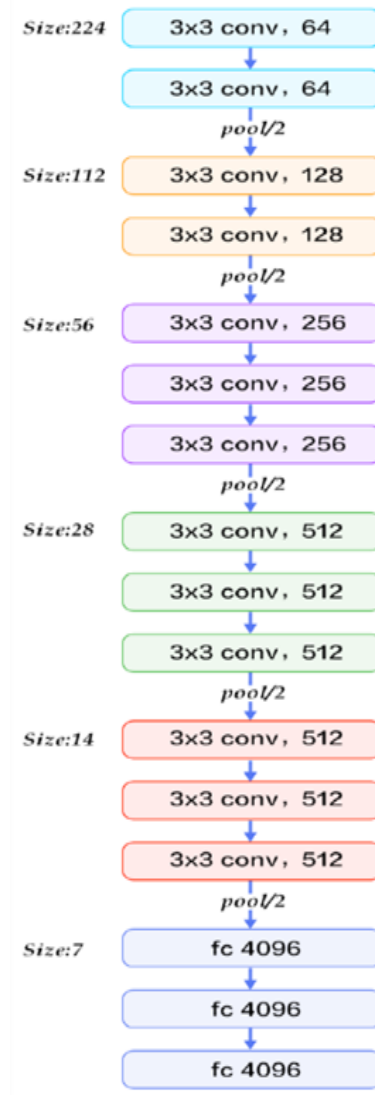
სურ. 14. LeNet-5 ქსელის არქიტექტურა ხელნაწერ სიმბოლოთა გამოსაცნობად [15]

AlexNet არქიტექტურა შეიცავს 5 კონვოლუციურ და 3 სრულადბმულ შრეს (სურ. 15).



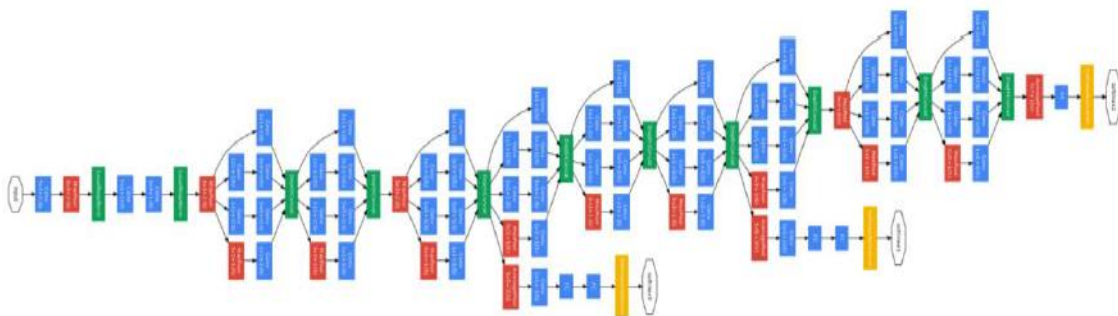
სურ. 15. AlexNet ქსელის არქიტექტურა [16]

მე-16 სურ-ზე ნაჩვენებია VGG-16 ქსელის არქიტექტურა



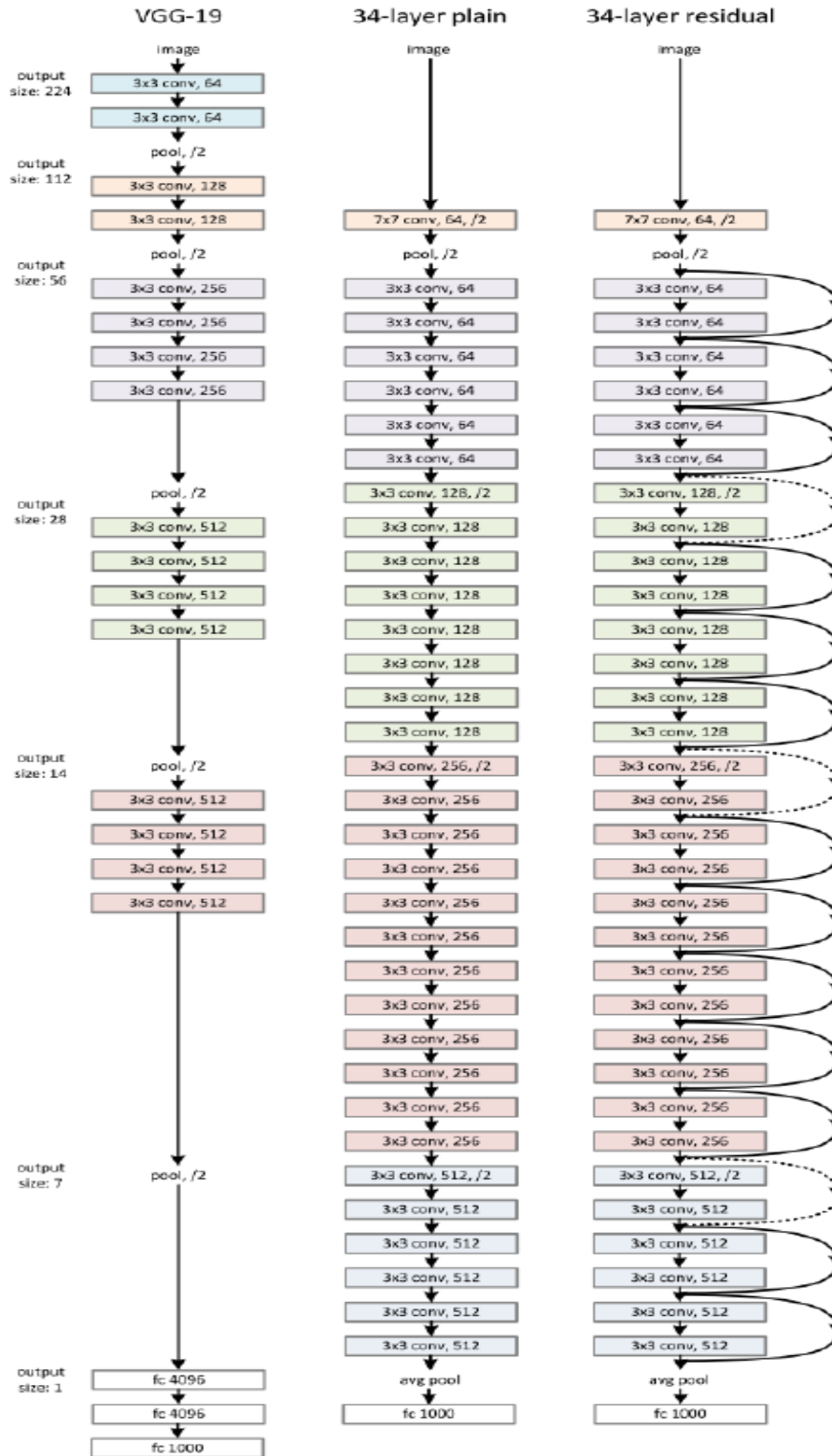
სურ. 16. VGG-16 ქსელის არქიტექტურა [17]

მე-17 სურ-ზე ნაჩვენებია GoogLeNet ქსელის არქიტექტურა



სურ. 17. GoogLeNet ქსელის არქიტექტურა [18]

მე-18 სურ-ზე ნაჩვენებია ResNet-34 ქსელის არქიტექტურა



სურ. 18. ResNet-34 ქსელის არქიტექტურა [19]

დასკვნა

კონვოლუციური (ხვეული) ნეირონული ქსელის გამოყენება ამოცანათა მრავალი ტიპის ამოსახსნელად შეიძლება: გამოსახულებათა საკლასიფიკაციოდ, მეტყველების გამოსაცნობად, ტექსტების დასამუშავებლად და ა.შ. ხვეული ნეირონული ქსელის გამოყენებისას შეიძლება დასასწავლ პარამეტრთა რაოდენობის არსებითად შემცირება და კლასიფიკაციის მაღალი ხარისხის მიღებაც.

კონვოლუციური ქსელების კიდევ ერთ უპირატესობას წარმოადგენს მათი უნივერსალობა გამოცნობის ამოცანებში: ნეირონული ქსელების ამ ნაირსახეობის გამოყენება შეიძლება პირთა ამოსაცნობადაც და კომპიუტერული ხედვის სისტემათა დასაპროექტებლადაც. ცნობილია ისიც, რომ კონვო-

ლუციური ქსელების გამოყენება სპეციალურ კლასიფიკატორთან ერთად ცალკეულ ხელნაწერ სიტყვათა ადვილად გარჩევის საშუალებასაც იძლევა.

დაბოლოს, ღირს იმის ხსენებაც, რომ კონვოლუციური ქსელი არის დამახინჯებათა შემცველი სიმბოლოების ამოცნობის ყველაზე საუკეთესო ალგორითმი. ასეთი გამოსახულებების დამუშავება წარმოადგენს ძალიან სერიოზულ ამოცანას, რომლის ეფექტური ამოხსნა სხვა ტიპის ნეირონული ქსელებით პრინციპულად შეუძლებელია. ამ ამოცანის გადასაჭრელად აუცილებელია გამოცნობის მრავალმოდულიანი სისტემის აგება, რომელიც იყენებს კონვოლუციურ ქსელებს, სპეციალურ კლასიფიკატორსა და გარდამსახველ ქსელს ცალკეულ სიმბოლოთა ამოსაცნობად გამოსახულებაზე.

ლიტერატურა

1. Haykin S. Neural networks: A full course. M.: "Williams". 2006. (in Russian).
2. Osovsky S. Neural networks for information processing. M.: "Finance and statistics". 2002. (in Russian).
3. Goodfellow I., Bengio Y., Courville A. Deep learning. MIT Press. 2016.
URL: <http://www.deeplearningbook.org>
4. Tarkov M. Electronic course "Neurocomputer systems".
URL: <http://www.intuit.ru/studies/courses/61/61/info> (in Russian).
5. Nielsen M. Neural networks and deep learning.
URL: <http://neuralnetworksanddeeplearning.com>
6. Krizhevsky A., Sutskever I., Hinton G. E. ImageNet classification with deep convolutional neural networks. Advances in neural information processing systems. 2012.
URL: <https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>
7. Backpropagation in convolutional neural network. URL: <https://www.slideshare.net/kuwajima/cnnbp>
8. Szegedy C., Vanhoucke V., Ioffe S., Shlens J. Rethinking the inception architecture for computer vision.
URL: <https://arxiv.org/pdf/1512.00567v3.pdf>
9. Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition.
URL: <https://arxiv.org/pdf/1409.1556.pdf>
10. Lin M., Chen Q., Yan S. Network in network. URL: <https://arxiv.org/pdf/1312.4400v3.pdf>
11. Long J., Shelhamer E., Darrell T. Fully convolutional networks for semantic segmentation. URL: <https://arxiv.org/pdf/1411.4038.pdf>
12. He K., Zhang X., Ren S., Sun J. Deep residual learning for image recognition.

- URL: <https://arxiv.org/pdf/1512.03385v1.pdf>
13. URL: <https://habr.com/en/company/mailru/blog/311706/> (in Russian).
 14. Caffe. URL: <http://caffe.berkeleyvision.org/>
 15. LeNet-5 – A classic CNN architecture. URL: <https://engmrk.com/lenet-5-a-classic-cnn-architecture/>
 16. Finetuning AlexNet with TensorFlow. URL: <https://kratzert.github.io/2017/02/24/finetuning-alexnet-with-tensorflow.html>
 17. Image classification. URL: https://github.com/PaddlePaddle/book/tree/develop/03.image_classification;
URL:
https://www.paddlepaddle.org.cn/documentation/docs/en/beginners_guide/basics/image_classification/index_en.html
 18. GoogLeNet in Keras. URL: <https://gist.github.com/joelouismarino/a2ede9ab3928f999575423b9887abd14>
 19. An overview of ResNet and its variants. URL: <https://towardsdatascience.com/an-overview-of-resnet-and-its-variants-5281e2f56035>

UDC 004.9

SCOPUS CODE 1705

Convolutional Neural Networks

- Archil Prangishvili** Department of Computer Engineering, Georgian Technical University, 77 M. Kostava str., 0160 Tbilisi, Georgia
E-mail: a_prangi@gtu.ge
- Oleg Namicheishvili** Department of Computer Engineering, Georgian Technical University, 77 M. Kostava str., 0160 Tbilisi, Georgia
E-mail: o.namicheishvili@gtu.ge
- Mikhael Ramazashvili** Department of Computer Engineering, Georgian Technical University, 77 M. Kostava str., 0160 Tbilisi, Georgia
E-mail: m.ramazashvili@gtu.ge

Reviewers:

Z. Gasitashvili, Professor, Faculty of Informatics and Control Systems, GTU

E-mail: zur_gas@gtu.ge

N. Mchedlisvili, Professor, Faculty of Informatics and Control Systems, GTU

E-mail: galoba47@mail.ru

Abstract. In the aspect of deep learning, the article considers the so-called convolutional neural networks. In particular, the following questions are systematically stated in the work: the “convolution” operation, the general structure of the convolutional layer, the input and output data of the convolutional network, the method of back propagation of error for convolutional neural networks, the determination of the number of trained parameters, the estimation of the amount of memory needed to store the network, an example of the use of convolutional neural networks in the problem of classifying handwritten numbers, the principles of constructing convolutional networks, the degradation of the model, deep residual networks and an example of applying nets in the classification of handwritten numbers. Special attention is focused on the development of convolutional neural network architectures.

Key words: Convolution operation; deep residual network; error back propagation method; handwritten digit classification; learning parameters; model degradation..

UDC 004.9

SCOPUS CODE 1705

Сверточные нейронные сети

- Арчил Прангишвили** Департамент компьютерной инженерии, Грузинский технический университет, Грузия, 0160, Тбилиси, ул. М. Костава, 77
E-mail: a_prangi@gtu.ge
- Олег Намичейшвили** Департамент компьютерной инженерии, Грузинский технический университет, Грузия, 0160, Тбилиси, ул. М. Костава, 77
E-mail: o.namicheishvili@gtu.ge
- Михаил Рамазашвили** Департамент компьютерной инженерии, Грузинский технический университет, Грузия, 0160, Тбилиси, ул. М. Костава, 77
E-mail: m.ramazashvili@gtu.ge

Рецензенты:

З. Гаситашвили, профессор факультета информатики и систем управления ГТУ

E-mail: zur_gas@gtu.ge

Н. Мchedlishvili, профессор факультета информатики и систем управления ГТУ

E-mail: galoba47@mail.ru

Аннотация. В аспекте глубокого обучения статья рассматривает так называемые свёрточные нейронные сети. В частности в работе систематически изложены следующие вопросы: операция «свертки», общая структура свёрточного слоя, входные и выходные данные свёрточной сети, метод обратного распространения ошибки для свёрточных нейронных сетей, определение количества обучаемых параметров, оценка объема памяти, необходимого для хранения сети, пример применения свёрточных нейронных сетей в задаче классификации рукописных цифр, принципы построения свёрточных сетей, проблема **деградации** модели, глубокие остаточные сети и пример применения глубоких остаточных сетей в задаче классификации рукописных цифр. Специально фокусируется внимание на развитии архитектур свёрточных нейронных сетей.

Ключевые слова: операция свёртки; метод обратного распространения ошибки; обучаемые параметры; классификация рукописных цифр; деградация модели; глубокая остаточная сеть.

განხილვის თარიღი 27.01.2020

შემოსვლის თარიღი 21.04.2020

ხელმოწერილია დასაბეჭდად 29.09.2020