

## **СРАВНИТЕЛЬНЫЙ АНАЛИЗ ФУНКЦИЙ АКТИВАЦИИ И ИХ ВЛИЯНИЕ НА ТОЧНОСТЬ РЕ-ИДЕНТИФИКАЦИИ ЛЮДЕЙ С ИСПОЛЬЗОВАНИЕМ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ**

*С. А. Игнатьева*

*Полоцкий государственный университет, Полоцк*

Ре-идентификация человека представляет собой процесс идентификации человека в другом месте или в другое время по видеоданным распределенных систем видеонаблюдения, и сопровождается рядом таких проблем как степень освещенности, окклюзии, разное качество изображений, похожесть разных людей, и непохожесть одного человека с разных углов обзора. На вход алгоритму ре-идентификации подаются изображения людей и требуется определить, как изображения-запросы соотносятся с изображениями, имеющимися в галерее набора данных. Строится дескриптор изображения-запроса, а затем осуществляется поиск по базе признаков, извлеченных для каждого изображения человека. Наиболее эффективным методом извлечения признаков на изображении является применение сверточных нейронных сетей (СНС).

Признаки, извлекаемые нейронной сетью (НС), влияют на эффективность работы алгоритма ре-идентификации и определяются в процессе обучения. Зависят от многих факторов: обучающая выборка, скорость и количество эпох обучения, архитектура используемой СНС и модификации ее структуры. Например, для повышения эффективности работы алгоритма и улучшения динамики обучения можно использовать разные функции активации, и исследования показывают, что для разных задач компьютерного зрения наилучшие показатели могут оказаться у разных функций. В таблице 1 представлены некоторые из них, используемые в глубоких СНС и указывается для каких задач функция была предложена и показала наибольшую эффективность.

Одной из наиболее распространенных функций активации в настоящее время является функция активации ReLU, основным преимуществом которой является низкая вычислительная сложность: при прямом проходе отрицательные значения приравниваются к 0, положительные остаются неизменными; при обратном проходе производная равна 0 для отрицательных значений, и 1 для положительных. При использовании этой функции активации можно столкнуться с проблемой взрывных градиентов, когда веса сети могут начать быстро увеличиваться. Так же для ReLU характерна такая особенность, как прореживание нейронов, т. е. нейроны, которые не активированы изначально, никогда не смогут быть активированными, и к ним будут добавляться те нейроны, на входы которых поступали отрицательные значения. Алгоритм градиентного спуска не сможет настраивать веса таких нейронов, и, с одной стороны, это облегчает сеть, так как прореживает ее и обучение осуществляется быстрее, с другой – приводит к потере части информации.

Таблица 1 – Функции активации и их область применения

Функция активации	Формулы и коэффициенты	Лучшие результаты
ReLU	$\varphi(x) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases}$	
Leaky-ReLU	$\varphi(x) = \begin{cases} x, & x > 0 \\ \alpha x, & x \leq 0 \end{cases}, \text{ где } \alpha = 0.01$	В глубоких НС при сравнении с ReLU и tanh.
PReLU	$\varphi(x) = \begin{cases} x, & x > 0 \\ \alpha x, & x \leq 0 \end{cases}, \text{ где } \alpha \text{ настраивается в процессе обучения}$	В сравнении с Leaky-ReLU и ReLU для классификации изображений на CIFAR-10 и CIFAR-100.
RReLU	$\varphi(x) = \begin{cases} x, & x > 0 \\ \alpha x, & x \leq 0 \end{cases}, \text{ где } \alpha \text{ принимает случайное значение на каждом слое}$	
ELU	$\varphi(x) = \begin{cases} x, & x \geq 0 \\ \alpha(e^x - 1), & x < 0 \end{cases}$	В задаче классификации изображений по сравнению с ReLU, Leaky-ReLU, SReLU на MNIST и CIFAR-10/100
SELU	$\varphi(x) = \lambda \begin{cases} x, & x \geq 0 \\ \alpha(e^x - 1), & x < 0 \end{cases}, \text{ где } \alpha = 1.67326, \lambda = 1.0507$	Для MNIST и CIFAR10 в задаче классификации, по сравнению с НС с пакетной нормализацией.
GELU	$\varphi(x) = x \cdot \frac{1}{2} \left[ 1 + \operatorname{erf} \left( \frac{x}{\sqrt{2}} \right) \right] \approx$ $\approx 0,5x(1 + \tanh(\sqrt{2/\pi}(x + 0.044715x^3)))$ или $\varphi(x) = x\sigma(1.702x), \text{ где}$ $\sigma(x) = 1/(1 + e^{-x})$	По сравнению с ReLU и ELU в задаче распознавания речи на TIMIT; классификации изображений CIFAR-10/100, MNIST.
Swish	$\varphi(x) = x\sigma(\beta x)$	В задаче классификации изображений по сравнению с Leaky-ReLU, PReLU, ELU, SELU, GELU на CIFAR-10/100, ImageNet; в задаче машинного перевода на WMT2014 English-German
Mish	$\varphi(x) = x \tanh(\operatorname{softplus}(x)) =$ $= x \tanh(\ln(1 + e^x))$	В задаче классификации изображений по сравнению с Leaky-ReLU, PReLU, ELU, SELU, GELU, Swish, SReLU на CIFAR-10, для 11 разных НС; для задачи обнаружения объектов в YOLOv4

Решением этой проблемы может быть функция активации Leaky-ReLU, которая оставляет небольшие отрицательные значения нейронов и сохраняет за ними возможность быть активированными. Это достигается за счет коэффициента  $\alpha$ , который принимает небольшие значения, обычно 0.01. Вычислительная сложность так же незначительна, а при обратном проходе использование углового коэффициента позволяет добиться ненулевых градиентов при отрицательных значениях, что позволяет весам обновляться.

В [1] проводится исследование, направленное на анализ влияния угла наклона отрицательной части функции на задаче классификации изображений, и рассматриваются такие модификации, как PReLU – Параметрическая ReLU, и

RReLU – рандомизированная ReLU. В PReLU оптимальное значение коэффициента  $\alpha$  подбирается в процессе обучения нейронной сети, в RReLU –  $\alpha$  задается случайным образом для каждого слоя. Исследование показало, что PReLU показывает лучшие результаты, однако при обучении на мелкомасштабном наборе данных сеть с PReLU склонна к переобучению, и функция RReLU оказывается более эффективной.

Существуют и другие модификации ReLU, например ELU, SeLU, GeLU, которые имеют небольшой наклон в отрицательной части области определения. В функции активации ELU для отрицательных значений используется логарифмическая кривая. Такой подход позволяет достигнуть насыщения нейронов в этой области, и тем самым уменьшить вариативность данных, распространяемых по сети. К недостаткам можно отнести большую вычислительную сложность, чем ReLU или Leaky-ReLU.

Функция SELU является масштабированным вариантом ELU, и в [2] приводятся ряд теорем, которые обосновывают и определяют значения для коэффициентов  $\alpha$  и  $\lambda$  и устанавливают их равными 1.67326 и 1.0507 соответственно. Функция SELU обладает эффектом самонормализации, т. е. все выходы после функции активации имеют среднее значение равное 0 и стандартное отклонение равное 1. Это обеспечивает сети более быструю сходимость. При использовании этой функции активации следует использовать начальную инициализацию весов, соответствующую нормальному распределению, для обеспечения самонормализующих свойств. Самонормализация позволяет избежать взрывных и исчезающих градиентов.

Основным отличием функции GELU от рассмотренных выше модификаций ReLU является то, что GELU является невыпуклой немонотонной функцией, нелинейна в положительной области и имеет кривизну во всех точках. Увеличенная кривизна и немонотонность позволяют GELU легче аппроксимировать сложные функции, и в отличие от ReLU, GELU не блокирует вход в зависимости от его знака.

В [3] для поиска наилучшей функции активации использовалась автоматическая генерация, основанная на последовательном переборе унарных и бинарных функций, которые поочередно объединяются, а результат оценивается эмпирически. Таким образом авторы исследования обнаруживают функцию Swish. Как видно из таблицы 1, Swish похожа на GELU и отличается лишь коэффициентом  $\beta$ , который может варьироваться. Изменяя  $\beta$ , можно регулировать свойства функции, чтобы максимизировать распространение информации и добиться плавных градиентов, делая обобщение лучше и быстрее. Swish не ограничена сверху, что позволяет избежать медленного обучения при почти нулевых градиентах, и ограничена снизу, что выгодно из-за сильной регуляризации, т. к. отбрасывается часть отрицательных входных данных.

Функция активации Mish, график которой имеет сходство со Swish и GELU, обладает всеми теми же свойствами, т.е. ее плавность позволяет оптимизировать градиентный поток, и она является саморегулируемой функцией. В ряде тестов, проведенных авторами этой функции активации, она показала лучшие результаты, чем другие рассмотренные ими функции.

Однозначно предсказать, как поведет себя та или иная функция активации для решения конкретной задачи компьютерного зрения, довольно сложно, и

поиск наиболее подходящей функции активации зачастую осуществляется экспериментально. Поэтому проведено исследование различных функций для решения задачи ре-идентификации человека на кадрах, полученных с камер видеонаблюдения.

Для тестирования алгоритма ре-идентификации с различными функциями активации использовался алгоритм, предложенный в [4], и рассматривались такие архитектуры НС как ResNet-50 и DenseNet-121, которые обучались в течение 60 эпох со скоростью 0,03 на наборе данных Market-1501. Результаты исследований приведены в таблице 2.

Таблица 2 – Результаты исследования влияния функции активации на точность ре-идентификации

Функция активации	ResNet-50				DenseNet-121			
	Потери	Время обучения	Точность ре-идентификации		Потери	Время обучения	Точность ре-идентификации	
ReLU	0,0057	86м 55с	R1: mAP:	0,81235 2 0,57311 5	0,0059	92м 6с	R1: mAP:	0,796318 0,566226
Leaky- ReLU	0,0057	86м 50с	R1: mAP:	0,81502 4 0,57444 1	0,0055	95м 52с	R1: mAP:	0,798458 0,566796
PReLU	0,0058	102м 49с	R1: mAP:	0,79572 4 0,55537 4	0,0059	138м 25с	R1: mAP:	0,781176 0,538241
RReLU	0,0049	90м 51с	R1: mAP:	0,80997 6 <b>0,582398</b>	0,0064	101м 57с	R1: mAP:	0,793646 0,566803
ELU	0,0059	85м 26с	R1: mAP:	0,79513 1 0,56174 0	0,0129	99м 54с	R1: mAP:	0,735154 0,490914
SELU	0,0083	86м 21с	R1: mAP:	0,77464 4 0,51913 4	0,0309	95м 12с	R1: mAP:	0,665677 0,407324
GELU	0,0059	88м 8с	R1: mAP:	<b>0,815915</b> 0,57923 5	0,0052	95м 13с	R1: mAP:	0,796912 0,569835
Swish	0,0048	104м 42с	R1: mAP:	0,81383 6 0,57681 9	0,0053	128м 57с	R1: mAP:	<b>0,800772</b> <b>0,568670</b>
Mish	0,0056	104м 52с	R1: mAP:	0,80849 2 <b>0,582007</b>	0,0052	130м 13с	R1: mAP:	0,787411 0,551814

Результаты исследований показали, что среди рассмотренных функций активации положительное влияние на точность ре-идентификации оказали такие функции как RReLU, GELU, Swish и Mish.

### Список литературы

1. Xu, B. Empirical Evaluation of rectified activations in convolutional network / B. Xu, N. Wang, T. Chen, M. Li [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1505.00853>. – Дата доступа: 18.10.2021
2. Klambauer, G. Self-Normalizing neural networks / G. Klambauer, T. Unterthiner, A. May, S. Hochreiter [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1706.02515v5>. – Дата доступа: 18.10.2021
3. Ramachandran, P. Swish: a Self-Gated activation function / P. Ramachandran, B. Zoph, Q.V. Le [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1710.05941v2> – Дата доступа: 18.10.2021
4. Person reID baseline pytorch [Электронный ресурс]. – Режим доступа: [https://github.com/layumi/Person\\_reID\\_baseline\\_pytorch](https://github.com/layumi/Person_reID_baseline_pytorch) – Дата доступа: 18.10.2021.

УДК 656.135

## РЕАЛИЗАЦИЯ РЕГИСТРАЦИИ ЗАЯВОК ПАССАЖИРОВ ТРАНСПОРТНОЙ СИСТЕМЫ

А. А. Левчук и И. С. Луковец

*Брестский государственный технический университет, Брест*

Научный руководитель: Шуть Василий Николаевич

*доцент БрГТУ, кандидат технических наук*

В настоящее время маршрутный транспорт получил широкое распространение среди граждан Республики Беларусь. В связи с этим, увеличилось количество машин на маршрутах; выезжая на маршрут водитель не понимает, будут ли пассажиры на пути его следования.

Нами разрабатывается клиентская и серверная составляющая системы перевозки пассажиров по запросу. В приложении для определения местоположения пользователя планируется использовать qr-код. QR-код, расположенный на каждой остановке общественного транспорта позволяет определить местоположение пассажира и название остановки. Для получения данной информации необходимо считать и расшифровать QR-код. Для обработки QR-кода удобно использовать готовые библиотеки. Для отбора таких библиотек был выполнен их анализ.

Для анализа использованы общедоступные библиотеки Mobile Vision API и Zxing.

В результате анализа была отобрана многоформатная библиотека Zxing для обработки изображений 1D/2D штрих-кодов с открытым исходным кодом [1]. Библиотека реализована на языке Java и может быть использована для различных платформ.

Обработка QR-кода с использованием Zxing в нашем приложении включает следующие действия :

- 1) Вызов приложения для сканирования qr- кода.
- 2) Получение входных данных:
  - а) наведение камеры на qr-код
  - б) вырезание qr-кода из общего изображения
- 3) Обработка входных данных:

Шаг 1. Чтение 5 бит системной информации.

Шаг 2. Маска для системной информации.

Шаг 3. Чтение заголовка данных.